Comparative genomics of microsatellite abundance: a critical analysis of methods and definitions

A thesis submitted in partial fulfilment of the

requirements for the degree of

PhD in Cellular and Molecular Biology

at the University of Canterbury

by

Iris Miriam Vargas Jentzsch

University of Canterbury

2009

Summary

This PhD dissertation is focused on short tandemly repeated nucleotide patterns which occur extremely often across DNA sequences, called microsatellites. The main characteristic of microsatellites, and probably the reason why they are so abundant across genomes, is the extremely high frequency of specific replication errors occurring within their sequences, which usually cause addition or deletion of one or more complete tandem repeat units. Due to these errors, frequent fluctuations in the number of repetitive units can be observed among cellular and organismal generations. The molecular mechanisms as well as the consequences of these microsatellite mutations, both, on a generational as well as on an evolutionary scale, have sparked debate and controversy among the scientific community. Furthermore, the bioinformatic approaches used to study microsatellites and the ways microsatellites are referred to in the general literature are often not rigurous, leading to misinterpretations and inconsistencies among studies. As an introduction to this complex topic, in Chapter I I present a review of the knowledge accumulated on microsatellites during the past two decades. A major part of this chapter has been published in the Encyclopedia of Life Sciences in a Chapter about microsatellite evolution (see Publication 1 in Appendix II).

The ongoing controversy about the rates and patterns of microsatellite mutation was evident to me since before starting this PhD thesis. However, the subtler problems inherent to the computational analyses of microsatellites within genomes only became apparent when retrieving information on microsatellite distribution and abundance for the design of comparative genomic analyses. There are numerous publications analyzing the microsatellite content of genomes but, in most cases, the results presented can neither be reliably compared nor reproduced, mainly due to the lack of details on the microsatellite search process (particularly the program's algorithm and the search parameters used) and because the results are expressed in terms that are relative to the search process (i.e. measures based on the absolute number of microsatellites). Therefore, in Chapter II I present a critical review of all available software tools designed to scan DNA sequences for microsatellites. My aim in undertaking this review was to assess the comparability of search results among microsatellite programs, and to identify the programs most suitable for the generation of microsatellite datasets for a thorough and reproducible comparative analysis of microsatellite content among genomic sequences. Using sequence data where the number and types of microsatellites were empirical know I compared the ability of 19

programs to accurately identify and report microsatellites. I then chose the two programs which, based on the algorithm and its parameters as well as the output informativity, offered the information most suitable for biological interpretation, while also reflecting as close as possible the microsatellite content of the test files.

From the analysis of microsatellite search results generated by the various programs available, it became apparent that the program's search parameters, which are specified by the user in order to define the microsatellite characteristics to the program, influence dramatically the resulting datasets. This is especially true for programs suited to allow imperfections within tandem repeats, because imperfect repetitions can not be defined accurately as is the case for perfect ones, and because several different algorithms have been proposed to address this problem. The detection of approximate microsatellites is, however, essential for the study of microsatellite evolution and for comparative analyses based on microsatellites. It is now well accepted that small deviations from perfect tandem repeat structure are common within microsatellites and larger repeats, and a number of different algorithms have been developed to confront the challenge of finding and registering microsatellites with all expectable kinds of imperfection. However, biologists have still to apply these tools to their full potential. In biological analyses single tandem repeat hits are consistently interpreted as isolated and independent repeats. This interpretation also depends on the search strategy used to report the microsatellites in DNA sequences and, therefore, I was particularly interested in the capacity of repeat finding programs to report imperfect microsatellites allowing interpretations that are useful in a biological sense. After analzying a series of tandem repeat finding programs I optimized my microsatellite searches to yield the best possible datasets for assessing and comparing the degree of imperfection of microsatellites among different genomes (Chapter III)

During the program comparisons performed in Chapter II, I show that the most critical search parameter influencing microsatellite search results is the minimum length threshold. Biologically speaking, there is no consensus with respect to the minimum length, beyond which a short tandem repeat is expected to become prone to microsatellite-like mutations. Usually, a single absolute value of ~12 nucleotides is assigned irrespective of motif length.. In other cases thresholds are assigned in terms of number of repeat units (i.e. 3 to 5 repeats or more), which are better applied individually for each motif. The variation in these thresholds is considerable and not always justifiable. In addition, any current minimum length measures are likely naïve because it is clear that different microsatellite motifs undergo replication slippage at different length thresholds. Therefore, in Chapter III, I apply

two probabilistic models to predict the minimum length at which microsatellites of varying motif types become overrepresented in different genomes based on the individual oligonucleotide frequency data of these genomes.

Finally, after a range of optimizations and critical analyses, I performed a preliminary analysis of microsatellite abundance among 24 high quality complete eukaryotic genomes, including also 8 prokaryotic and 5 archaeal genomes for contrast. The availability of the methodologies and the microsatellite datasets generated in this project will allow informed formulation of questions for more specific genome research, either about microsatellites, or about other genomic features microsatellites could influence. These datasets are what I would have needed at the beginning of my PhD to support my experimental design, and are essential for the adequate data interpretation of microsatellite data in the context of the major evolutionary units; chromosomes and genomes.

Acknowledgements

First of all, I would like to thank my supervisor, Professor Neil J. Gemmell for opening the doors of his lab to me, and for his exceptional optimism and patience. Thanks to Neil's multiple project involvements I had first hand access to the University of Canterbury Supercomputer (BlueFern), and to the BestGrid Scientific Network. My PhD was co-funded by the Marsden Fund and a University of Canterbury Doctoral Scholarship.

Special thanks to the BlueFern team Vladimir Mencl, Colin McMurtrie and Tony Dale for their help with basic software problems and for their enthusiastic interest in research.

Thanks to Lisha Naduvilezhath for being an awesome co-worker, for teaching me java programming, for not laughing about my programs, and for being such a sweet and loving friend.

And talking about sweet and loving friends, I need to insert here a long list of nice people I met in New Zealand. However, dear reader, please don't feel offended if your name is not in this list. My last days in New Zealand have been so stressy that parts of my brain would randomly fall asleep! This list may therefore be biased towards those who lastly cheered me up or fed me. Here goes the list: Margee Will, Andrew Bagshaw, Andrea Kutinova (turned Menclova) and Vladimir Mencl, Maggie Tisch, Sandra Gandre, Wiebke Müller, Tammy Steeves, Jonci Wolff, Dan White, Andrea Contreras, Moffat Mathews, etc!

To Thorsten, Margee, and Andrew: it was great fun sharing the nightshift with you!

Thanks to Dr. Geoff Buckett for his serene wisdom and dedication.

Thanks to Jo (Blueberry) and Paul, and their flatmates, for allowing me to live in their flat once I completely run out of money.

And thank you to the anonymous coffee lover who donated his/her Breville espresso machine for common use in the tea room:

A heartfelt thank you to the Horn Family; Ingrid, Margot, Bernd, and Carsten, for their emotional and financial support during the last year of my PhD. Special thanks to Carsten for writing a program (IrSa) for me!

Thanks to my family: Ursula, Raul, Paul and Andreas for being always in contact and for taking good care of themselves. Special thanks to my mom who took great care of my beloved dog Konrad after I disappeared to do my PhD at the other end of the world.

Finally, the most important thank you to Thorsten Horn for his amazing patience and his emotional, inspirational, musical, critical, and financial support. Your love is the best that ever happened to me ☺!!!

Thank you to all the kiwis I met, and the ones I did not meet, for making New Zealand such a relaxing and friendly place to live.

Table of Contents

List of Figure Captions

List of Table Captions

Statement of Sources

Genomic sequences

Complete genome sequences listed in **tables S1** and **S2** were downloaded from the NCBI (ftp://ftp.ncbi.nih.gov/genomes/) via the biomirror interface (http://www.biomirror.org.nz/), and from the UCSC Genome Browser. Additionally, the yeast genome (last modified on 30.11.2006) was downloaded from the *Saccharomyces* Genome Database (http://www.yeastgenome.org/).

Table S1: Eukaryotic genome sequences downloaded from NCBI

Species	Scientific name	Assembly name (based on UCSC)	Publication date	Alternative Assembly name
Human	*Homo sapiens*	Hg18	March 2006	NCBI 36.1
		Hg17	April 2004	NCBI 35
		Human Celera		
Chimp	*Pan troglodytes*	panTro2	March 2006	Pan_troglodytes-2.1
Rhesus monkey	*Macaca mulatta*		December 2005	Mmul_051212
Dog	*Canis familiares*	Canfam2	May 2005	__
		Canfam1	July 2004	__
Mouse	*Mus musculus*	mm8	February 2006	__
		Mouse Celera		__
Rat	*Ratus norvergicus*	rn4	November 2004	__
		Rat Celera		__
Cow	*Bos taurus*	bosTau2	October 2005	Btau 2.0
Chicken	*Gallus gallus*	galGal3	May 2006	Gallus_gallus-2.1
Opossum	*Monodelphis domestica*	monDom4	January 2006	
Platypus	*Ornithorhynchus anatinus*	ornAna1	March 2007	Ornithorhynchus_anatinus-5.0.1
Zebrafish	*Danio rerio*	danRer4	March 2006	Zv6
		danRer2	June 2004	
Honeybee	*Apis mellifera*	apiMel2	January 2005	
Red flour beetle	*Tribolium castaneum*	__	September 2005	Tcas_2.0
Roundworm	*Caenorhabditis elegans*	ce2	March 2004	WS120
Yeast	*Saccharomyces cerevisiae*	__	__	__
Rice	*Oryza sativa* (*japonica* cultivar-group)	__	__	__
Arabidopsis	*Arabidopsis thaliana*	__	__	__
Plasmodium	*Plasmodium falciparum* 3D7	__	__	__

Table S2: Archaea and Bacteria genomic sequences downloaded from NCBI

Species/Scientific name	Accession number	Publication date	Update date
Hyperthermus butylicus DSM 5456	NC_008818	Jan 23 2007	Dec 12 2007
Methanocaldococcus jannaschii DSM 2661	NC_00171732 NC_001732 NC_000909	Aug 21 1996 Aug 21 1996 Sep 10 2001	Dec 3 2007 Dec 3 2007 Dec 3 2007
Natronomonas pharaonis DSM 2160 and plasmids PL 131 and PL233	NC_007426 NC_007427 NC_007428	Oct 3 2005 Oct 3 2005 Oct 3 2005	Dec 12 2007 Dec 12 2007 Oct 7 2005
Pyrobaculum aerophilum str. IM2	NC_003364	Dec 12 2001	Dec 2 2007
*Methanosaeta thermophila*_PT	NC_008553	Oct 27 2006	Dec 6 2007
Neisseria meningitidis FAM18	NC_008767	Jan 10 2007	Dec 4 2007
Brucella melitensis biovar *Abortus* 2308	NC_007618 NC_007624	Nov 28 2005 Dec 5 2005	Dec 7 2007 Dec 7 2007
Bacillus anthracis str. Ames	NC_003997	May 16 2002	Dec 4 2007
Escherichia coli K12	NC_000913	Oct 15 2001	Dec 20 2007
Campylobacter jejuni subsp. *jejuni* NCTC 11168	NC_002163	Sept 27 2001	Dec 2 2007
Mycobacterium tuberculosis H37Rv	NC_000962	Sep 7 2001	Nov 29 2007
Bacillus thuringiensis str. Al Hakam	NC_008598 NC_008600	Nov 29 2006	Dec 12 2007
Clostridium tetani E88	NC_004565	Feb 13 2003	Jun 4 2007
Lactobacillus casei ATCC 334	NC_008502	Oct 21 2006	Oct 31 2006

Table S3: Eukaryotic genome sequences downloaded from the UCSC Genome Browser FTP site

Species	Scientific name	Assembly name (based on UCSC)	Publication date	Alternative Assembly name
Medaka	*Oryzias latipes*	oryLat1	April 2006	
Stickleback	*Gasterosteus aculeatus*	gasAcu1	February 2006	
Fruitfly	*Drosophila melanogaster*	dm2	April 2004	
Mosquito	*Anopheles gambiae str.* PEST	anoGam1	February 2003	MOZ2
Pufferfish	*Tetraodon nigroviridis*	tetNig1	February 2004	V7
Horse	*Equus caballus*	equCab1	January 2007	

Bioinformatic toolkits

- Genome Browser and Table Browser from the University of Santa Cruz in California: http://www.genome.ucsc.edu/
- Galaxy: http://main.g2.bx.psu.edu/

Programs

Table S4: Repeat finding programs

Year	Program	Language	Webpage	Publication
1997	**Tandyman**	Perl	http://hemisphere.lanl.gov/tandyman/cgi-bin/tandyman.cgi	*NP (LEACH and CLELAND 1997)
1999	Tandem Repeat Finder (**TRF 4.00**)	C	http://tandem.bu.edu/trf/trf.html	(BENSON 1999)
2000	**SSR screener**	C	ftp://ftp.technion.ac.il/pub/supported/biotech/	*NP (GUR-ARIE *et al.* 2000)
2001	**SSRIT** Simple Sequence Repeat Identification Tool	Perl	http://www.gramene.org/db/searches/ssrtool	*NP (TEMNYKH *et al.* 2001)
~2002	MIcroSAtellite identification tool (**MISA**)	Perl	http://pgrc.ipk-gatersleben.de/misa/	*NP Author: Thomas Thiel (THIEL *et al.* 2003)
2002	**ComplexTR**	__	http://malawimonas.bcm.umontreal.ca:8091/anabench/Anabench-Jsp/Applications/ListApplications.jsp	(HAUTH and JOSEPH 2002)
2002	Tandem Repeat Occurrence Locator (**TROLL**)	__	http://finder.sourceforge.net/ http://al.jalix.org/FORRepeats/ (but the link seems to be broken)	(CASTELO *et al.* 2002)
2003	**Sputnik II**	C	http://wheat.pw.usda.gov/ITMI/EST-SSR/LaRota/	*NP (LA ROTA *et al.* 2005)
2003	Search for Tandem Repeats IN Genomes (**STRING**)	C, java	http://bioinf.dms.med.uniroma1.it/JSTRING/	(PARISI *et al.* 2003)
2003	**Poly**	Python	http://www.bioinformatics.org/poly/wiki/	(BIZZARO and MARX 2003)
2003	**Mreps** (ver 2.5)	C	http://bioinfo.lifl.fr/mreps/	(KOLPAKOV *et al.* 2003)
2003	**SSRfinder**	Perl	http://www.maizemap.org/bioinformatics/SSRFINDER/	*NP Author: Steven Schroeder, SchroederSG@missouri.edu

Year	Program	Language	Webpage	Publication
2003	Perfect Tandem Repeat Executable (**ptrfinder**)	C	http://ncisgi.ncifcrf.gov/~collinsj/Tandem_Repeats/downloads/	*NP Author: Jack R. Collins (COLLINS *et al.* 2003)
2004	Approximate Tandem Repeats hunter (**ATRhunter**)	java	http://bioinfo.cs.technion.ac.il/atrhunter/ATRHunter.htm	(WEXLER *et al.* 2005)
2004	Search for Tandem Approximate Repeats (**STAR**)	__	http://atgc.lirmm.fr/star/	(DELGRANGE and RIVALS 2004)
2004	Tandem Repeat Analyzer (**TRA** and **E-TRA**)	C++	ftp://ftp.akdeniz.edu.tr/Araclar/TRA/	Described very briefly in (BILGEN *et al.* 2004), (KARACA *et al.* 2005)
2005	**Msatfinder/Msatminer**	Perl	http://www.genomics.ceh.ac.uk/msatfinder/	*NP (THURSTON and FIELD 2005)
2006	**SSRscanner**	Perl	No web page	(ANWAR and KHAN 2006)
2006	**Phobos**	C++	http://www.ruhr-uni-bochum.de/spezzoo/cm/cm_phobos.htm	*NP But a complete user manual is available, which also explains how the program works (MAYER 2007)
2006	**FireμSat:**	__	http://www.dna-algo.co.za/	(DE RIDDER *et al.* 2006)
2007	Imperfect Microsatellite Extractor (**IMEx 1.0**)	C	http://bioinfo.lifl.fr/mreps/	(MUDUNURI and NAGARAJARAM 2007)
2007	Tandem Repeat Software (**TRED**)	C++	http://www.sci.brooklyn.cuny.edu/~sokol/tandem/	(SOKOL *et al.* 2007)
2007	**SciRoKo 3.3 SciRoKoCo**	C	http://www.kofler.or.at/Bioinformatics/SciRoKo/index.html	(KOFLER *et al.* 2007)
2007	**tandem**	__	http://www.cs.brown.edu/people/domanic/tandem/	(DOMANIC and PREPARATA 2007)

*NP : No publication was available describing the algorithm. Therefore I mention the authors and/or the application paper where the program was first used.

Accessory software tools

The manipulation, organization and comparison of microsatellite datasets was performed using Microsoft Visual Basic for Excel macros and Java programs written in conjunction with Lisha Naduvilezhath, an exchange student from the Wolfgang Goethe University in Germany.

Table S5: Accessory software tools

Program	**Author**	**Use**
Timer.exe	Harold Kaplan http://www.toad.net/~jkaplan2/timer/	To measure the execution time of command line programs in Windows
Cream (for Vim) ver 0.39	http://cream.sourceforge.net.digitect	Edition of large text files and scripts
Cygwin	Red Hat Inc. http://www.cygwin.com/	Linux-like environment for Windows
Eclipse SDK, Version: 3.3.1.1	Apache Software Foundation http://www.apache.org/	Java programming platform
R	http://www.r-project.org/	Language and environment for statistical computing
Split.pl	Vladimir Mencl	To split large datasets
Bash scripts	Iris Vargas Jentzsch	Scripts for control and combination of programs
Visual Basic for Excel Macros TRF RedundancyEliminator	Lisha Naduvilezhath Iris Vargas Jentzsch	The redundancy from TRF results was initially filtered using these macros.
Java Programs baseComposition BaseCount CountAs.java CountTs.java CountCs.java CountGs.java CountNs.java MononuclFreq.java DinuclFreq.java TrinuclFreq.java MononuclMarkov.java DinuclMarkovjava TrinuclMarkov.java onCoordinates Merge.java MergeContigs.java MsatDensity.java OverlappIntersection.java RandomFeatureOverlap.java SortAfterChromosomes.java TRFredundancyEliminator.java onLines CombineSputFile.java CombineTRFFile.java CountFAsequences.java DeleteFirstLines.java JoinTRFresults.java NameSequences.java Replace.java wublast MsatFilter.java MsatFilter_single.java	Lisha Naduvilezhath Iris Vargas Jentzsch	The programs in the baseComposition package were used to obtains nucleotide, dinucleotide, and trinucleotide counts for basic sequence characterization and for the calculation of expected microsatellite numbers in Chapter IV (*Freq.java and *Markov.java programs). The programs in the onCoordinates package were used for processing microsatellite search results from different programs, to calculate microsatellite coverage, and to eliminate redundancy from TRF results. The programs from the onLines package were used for combining search results from different programs. The wublast package contains the programs to filter microsatellite datasets from any program output based on minimum length or minimum number of repeats of the hits.
IrSa	Carsten Horn	Program to perform exhaustive searches of tandem repeats of given motifs in small DNA sequences.
Bash scripts	Iris Vargas Jentzsch	Scripts for control and combination of programs

Computing facilities

- Bioinformatics room from the Molecular Ecology Lab, University of Canterbury.
- University of Canterbury Supercomputer:

  http://www.ucsc.canterbury.ac.nz/userdocs.shtml

References

ANWAR, T., and A. U. KHAN, 2006 SSRscanner: a program for reporting distribution and exact location of simple sequence repeats. Bioinformation **1:** 89-91.

BENSON, G., 1999 Tandem repeats finder: a program to analyze DNA sequences. Nucleic Acids Res **27:** 573-580.

BILGEN, M., M. KARACA, A. N. ONUS and A. G. INCE, 2004 A software program combining sequence motif searches with keywords for finding repeats containing DNA sequences. Bioinformatics **20:** 3379-3386.

BIZZARO, J. W., and K. A. MARX, 2003 Poly: a quantitative analysis tool for simple sequence repeat (SSR) tracts in DNA. BMC Bioinformatics **4:** 22.

CASTELO, A. T., W. MARTINS and G. R. GAO, 2002 TROLL--tandem repeat occurrence locator. Bioinformatics **18:** 634-636.

COLLINS, J. R., R. M. STEPHENS, B. GOLD, B. LONG, M. DEAN *et al.*, 2003 An exhaustive DNA micro-satellite map of the human genome using high performance computing. Genomics **82:** 10-19.

DE RIDDER, C., D. KOURIE and B. WATSON, 2006 FireμSat: An algorithm to detect microsatellites in DNA, pp. in *Proceedings of the Prague Stringology Conference 2006*, Prague.

DELGRANGE, O., and E. RIVALS, 2004 STAR: an algorithm to Search for Tandem Approximate Repeats. Bioinformatics **20:** 2812-2820.

DOMANIC, N. O., and F. P. PREPARATA, 2007 A novel approach to the detection of genomic approximate tandem repeats in the Levenshtein metric. J Comput Biol **14:** 873-891.

GUR-ARIE, R., C. J. COHEN, Y. EITAN, L. SHELEF, E. M. HALLERMAN *et al.*, 2000 Simple sequence repeats in *Escherichia coli*: abundance, distribution, composition, and polymorphism. Genome Res **10:** 62-71.

HAUTH, A. M., and D. A. JOSEPH, 2002 Beyond tandem repeats: complex pattern structures and distant regions of similarity. Bioinformatics **18 Suppl 1:** S31-37.

KARACA, M., M. BILGEN, A. N. ONUS, A. G. INCE and S. Y. ELMASULU, 2005 Exact tandem repeats analyzer (E-TRA): a new program for DNA sequence mining. J Genet **84:** 49-54.

KOFLER, R., C. SCHLÖTTERER and T. LELLEY, 2007 SciRoKo: A new tool for whole genome microsatellite search and investigation. Bioinformatics **23:** 1683-1685.

KOLPAKOV, R., G. BANA and G. KUCHEROV, 2003 mreps: efficient and flexible detection of tandem repeats in DNA. Nucl. Acids Res. **31:** 3672-3678.

LA ROTA, M., R. KANTETY, J.-K. YU and M. SORRELLS, 2005 Nonrandom distribution and frequencies of genomic and EST-derived microsatellite markers in rice, wheat, and barley. BMC Genomics **6:** 23.

LEACH, R. W., and C. CLELAND, 1997 Tandyman, pp. Los Alamos National Laboratory, California, USA.

MAYER, C., 2007 Phobos Version 3.3.2. A tandem repeat search program, pp.

MUDUNURI, S. B., and H. A. NAGARAJARAM, 2007 IMEx: Imperfect Microsatellite Extractor. Bioinformatics **23:** 1181-1187.

PARISI, V., V. DE FONZO and F. ALUFFI-PENTINI, 2003 STRING: finding tandem repeats in DNA sequences. Bioinformatics **19:** 1733-1738.

SOKOL, D., G. BENSON and J. TOJEIRA, 2007 Tandem repeats over the edit distance. Bioinformatics **23:** e30-35.

TEMNYKH, S., G. DECLERCK, A. LUKASHOVA, L. LIPOVICH, S. CARTINHOUR *et al.*, 2001 Computational and experimental analysis of microsatellites in rice (*Oryza sativa* L.): frequency, length variation, transposon associations, and genetic marker potential. Genome Res **11:** 1441-1452.

THIEL, T., W. MICHALEK, R. K. VARSHNEY and A. GRANER, 2003 Exploiting EST databases for the development and characterization of gene-derived SSR-markers in barley (*Hordeum vulgare* L.). Theor Appl Genet **106:** 411-422.

THURSTON, M. I., and D. FIELD, 2005 Msatfinder: detection and characterisation of microsatellites, pp. CEH Oxford.

WEXLER, Y., Z. YAKHINI, Y. KASHI and D. GEIGER, 2005 Finding approximate tandem repeats in genomic sequences. J Comput Biol **12:** 928-942.

Abbreviations and Definitions

Abbreviations

ATRs= Approximate Tandem Repeats
CNVs= Copy Number Variations
ESTs= Expressed Sequence Tags
Indel= Abbreviated combination of **in**sertions and **del**etions
SNP= Single Nucleotide Polymorphism
UCSC= University of California in Santa Cruz
NCBI= National Center for Biotechnology Information
ABCC= Advanced Biomedical Computing Centre, NCI-Frederick, Frederick, MD, USA.
GRID= Genome Repeats Information Database
GPL= General Public Licence
GUI= Graphical User Interface
TIGR= The Institute for Genomic Research
VNTRs=Variable Number of Tandem Repeats
Chr= Chromosome
SSM= Slipped-Strand Mispairing as defined by {Levinson, 1987 #102}
DSB= Double Strand Break
cmd=command line (computer program interface)
nt=nucleotide
Mb= Mega base pairs

Definitions

Algorithm.- a list of detailed instructions for solving a particular problem.
Clustered mutations.- copies of a mutant allele which arise by premeiotic mutations in the germline.
Deletion.- loss of a part of DNA from a chromosome. It can be as small as a single nucleotide, or a gene, and as big as a chromosomal segment or an entire arm. In the case of microsatellites it involves mostly whole repeat units, either one or several.
Genome.- full set of DNA molecules constituting the genetic material of an organism.
Indel.- is a term used in genetics to denote insertions *or* deletions in a DNA sequence. It is a mutation class implying the occurrence of insertions and deletions.
Insertion.- in genetics it is a kind of mutation where one or more nucleotide pairs are added to a DNA sequence. In the case of microsatellites it refers to the addition of one or more repeat units
Interspersed repeats.- repeated DNA segments located in dispersed regions across a genome. These are also known as mobile elements or transposable elements.
Lexicographic order.- when applied to permutations of nucleotides to form microsatellite motifs, lexicographic order refers to the alfabetical ordering of the nucleotide symbols. For example, permutations of ATC in lexicographic order are: ACT, ATC, CAT, CTA, TAC, TCA.
Molecular marker.- a variation in the genetic material at a single locus, resulting from an alteration or mutation These variations are detected by DNA-based genotyping techniques.
Motif.- is the nucleotide string which is repeated in tandem in a minisatellite or microsatellite. This is also referred to as ***period*** in the literature.

Neutral mutation.- a mutation whose effect is not strong enough to permit selection for or against it.
Neutral sequence.- a stretch of DNA which does not carry out any essential function in the genome and, therefore, is free to mutate randomly without affecting the fitness of the individual. This is a general term which should be used with caution, because the condition of neutrality will depend on the circumstances (environment).
Perfect repeat.- used for tandem repeats in which all motifs are exact copies of each other.
Phenome.- from the greek verb *phainein*, to show, denotes all traits that can be seen (phenotypic traits) and used to describe an organism. Phenotypic traits are influenced by genetic as well as by environmental factors.
Polymorphism.- refers to the occurrence or two or more different forms of the same gene or genetic marker, each of these being present in at least 1% of the population.
Proteome.- the entire complement of proteins expressed by a genome at a given developmental stage and under specified conditions.
Sequence annotation.- the action of positioning known as well as predicted genetic features (e.g. DNA and RNA genes, pseudogenes, non-coding regions, repetitive DNA, etc) across a genomic sequence. Any feature that can be anchored to a sequence is an annotation.
Standing genetic variation.- allelic variation that is currently segregating within a population; as opposed to alleles that appear by new mutation events.
String.- a concatenation of characters
Substitution.- is the kind of mutation where a nucleotide is replaced by a different one in the DNA sequence. It is a ***transition*** when a purine is substituted by another purine (A→G, G→A), or a pyrimidine by another pyrimidine (T or U→C, C→T or U). And when the substitutios occur between a purine (A,G) and a pyrimidine (T/U, C), or viceversa, it is called a ***transversion***.

Disambiguation of terminology

Short or simple tandem repeats (STRs), simple sequence repeats (SSRs), simple sequence length polymorphisms (SSLPs) and variable number tandem repeats (VNTRs), all refer to microsatellite markers.
Repeat unit=period of the tandem repeat
Motif= AG, T, GAC, etc.
Motif type=mono-, di-, tri-, tetra-, pentanucleotide.
Motif length= 1, 2, 3, … nt.

CHAPTER I: An Introduction to Microsatellites

Abstract

Microsatellites are highly mutable tandemly repeated sequences that are ubiquitously distributed in prokaryotic and eukaryotic genomes. Microsatellites became the preferred molecular marker for a variety of applications under the basic assumption that they are selectively neutral. However, the simplicity of this assumption contrasts with the observed variability of mutation rates across microsatellite loci and with the increasing evidence supporting microsatellite functionality. The evolutionary importance of microsatellites is only recently being uncovered with the intense study of regulatory mechanisms of gene expression and the interaction among genomic structures.

In this chapter I summarize the knowledge that has accumulated about microsatellites in the past twenty five years. For this I focus on describing well proven characteristics while simultaneously exposing common misconceptions and unclear details about these enigmatic tandem repeats.

---

* This chapter is a modified version of Vargas *et al (*2008) Evolution of microsatellite DNA. *in* Encyclopedia of Life Sciences, John Wiley & Sons, Ltd: Chichester. www.els.net

1.1 Introduction

The terms 'microsatellites', simple sequence repeats (SSRs), or short tandem repeats (STRs), are usually associated with very polymorphic and, therefore, highly informative molecular markers. Indeed, during the nineties, the analysis of microsatellites constituted the most popular molecular marker technique for genotyping individuals and populations. A microsatellite consists of tandem repetitions of very short repeat units (called motifs) within nucleic acid sequences. The most valued and intriguing characteristic of microsatellites is their extraordinary polymorphism, which is attributable to the occurrence of replication errors at a very high rate within the repeated structures. Microsatellites are also highly abundant within DNA sequences, having been found at high frequencies in every genome studied to date (KATTI *et al.* 2001; MORGANTE *et al.* 2002; TOTH *et al.* 2000; VAN BELKUM *et al.* 1998). Being so mutable and abundant, the probability to find one or more microsatellite loci showing differences among individuals, even if these are siblings or parent-offspring pairs, is exceptionally high.

Once their unprecedented variation was discovered, biologists found an impressive amount of applications for microsatellites. For example, the use of microsatellites as molecular markers has revolutionized paternity testing (AGAPITO *et al.* 2008; ZAJC and SAMPSON 1996) and molecular identification techniques, for the confirmation of family pedigrees (OKADA and TAMATE 2000), studies of reproductive success (MCLEAN *et al.* 2008), as well as for forensic investigations (HOFF-OLSEN *et al.* 2001). The omnipresence of microsatellites within genomes became also very important for genetic mapping projects, for example for the genomes of human (WERNER *et al.* 1999b), mouse \(RHODES *et al.* 1998), dog (WERNER *et al.* 1999a), trout (GUYOMARD *et al.* 2006); wheat (RÖDER *et al.* 1998), cotton (GUO *et al.* 2007), etc. Indeed, the analysis of microsatellite markers has become so widespread that their use became customary across all biological and medical areas of study.

The reasons for microsatellite hypervariability, however, were not easy to grasp under the gene-centric view of evolution which dominated genetic research during the period between 1980 and ~1995. Therefore, microsatellites were believed to be selectively neutral, which would mean that they don't carry out any essential function in the genome and, therefore, they are free to mutate randomly without affecting the fitness of the individual. This assumption is central to the utilization of microsatellites as molecular markers for population genetics, a field where microsatellite analysis is an established practice (JARNE

and LAGODA 1996). Some examples include the study of differentiation of population substructure (BALLOUX and LUGON-MOULIN 2002; WALTER and EPPERSON 2004, Koskinen, 2000 #520), the assessment of genetic diversity within and between populations (DEKA *et al.* 1995; IRION *et al.* 2003; LI *et al.* 2004; SUN *et al.* 2008; TSUTSUI *et al.* 2000, Hara, 2007 #910; ZHANG *et al.* 2008), the analysis of mating systems (VIARD *et al.* 1996), and parasitological analyses (recent examples: AL-JAWABREH *et al.* 2008; reviewed in BARKER 2002; HAVRYLIUK *et al.* 2008; MLAMBO *et al.* 2007; MONTOYA *et al.* 2007),

Due to the assumption of neutrality, microsatellites are either used as neutral genetic markers, or otherwise excluded from genetic analyzes. Microsatellite mutation rates and patterns have been intensely studied in order to infer models to describe these, to be used in studies applying microsatellite markers (DI RIENZO *et al.* 1994; KRUGLYAK *et al.* 1998; NEFF *et al.* 1999; SAINUDIIN *et al.* 2004). However, studies reporting on microsatellite mutations tend to be biased towards longer and more mutable microsatellites (BROHEDE *et al.* 2002) because otherwise the direct observation of microsatellite mutations would become too unlikely, and would therefore reduce the strength of the analysis. Despite of this, or maybe because of this, the range of mutation rates and patters observed is highly heterogeneous, exhibiting differences between loci differing in length, motif sizes, motif nucleotide composition (BACHTROG *et al.* 2000; SCHLÖTTERER and TAUTZ 1992) , and between species (WEBSTER *et al.* 2002).

In general, the nature of microsatellite mutations and their relevance in genome evolution and functioning, or lack thereof, constitute an ongoing debate. This debate is carried out contemporarily to the widespread application of microsatellites as molecular markers, although with increasing carefulness so not to stumble with unexpected or unexplainable results.

1.2 Repetitive DNA

The term "repetitive DNA" is used to refer to DNA sequence segments that are present in multiple copies across a chromosome or genome. DNA repeats can occur in four different orientations with respect to each other in the DNA helix: direct, reverse, complement, and reverse complement (**figure 1.1**). Direct repeats are sequences repeated in the same order in the same strand. In a reverse repeat, the same sequence is repeated with the order of nucleotides reversed in the same DNA strand. In a complement repeat, the complementary sequence to the original repeat occurs on the same strand. A reverse complement is the

equivalent of a reverse repeat occurring in the opposite strand, so that the complementary nucleotides of each each nucleotide from the original repeat are represented in an inverse order on the original strand. Reverse complement repeats are also called palindromic repeats when the repeated units are adjacent to each other, and these can form hairpin structures when the DNA becomes single-stranded (HUANG *et al.* 1998).

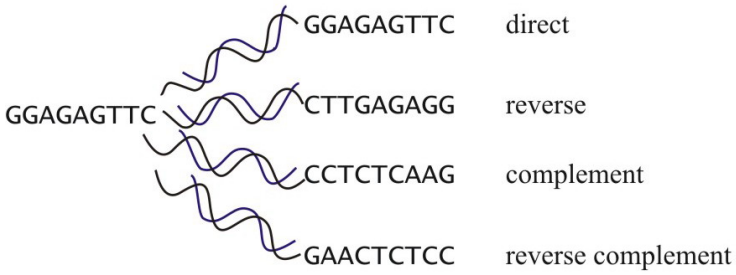

Figure 1.1: Kinds of repeats based on their orientation in the DNA strand

Direct repeats are by far the most abundant ones within genomes. From a chromosomal distribution point of view, these can be classified in two categories: genome-wide or interspersed repeats and tandem repeats. In the first case, individual repeat units are inserted apparently at random across the genome. In the later case the repeat units are arranged side by side forming a repeat array. This classification is, however, not mutually exclusive since each class retains characteristics of both (KAPITONOV *et al.* 2004).

Interspersed repeats are usually derived from transposable elements. These are relatively long DNA fragments (up to 20 -30 kb), initially alien to the genome in question, which can proliferate independently from the host genome, and insert themselves repeatedly in different (supposedly random) positions across the host genome. Many of these copies (most of them in the case of humans) have become inactive throughout time; they are incomplete or degenerated copies of the original transposable elements (JURKA *et al.* 2007)

Tandem repeats are, in general, smaller than interspersed repeats, but are also abundant and amply distributed throughout genomes. A sub-classification of tandem repeats is usually done based on the length of its repeat units (referred throughout this text as "motifs") and the specific position they occupy within a chromosome (KAPITONOV *et al.* 2004). It can occur that the number of tandem repetitions of a particular repeat locus varies among individuals: i.e. between parent and offspring, among the members of a population,

or even among cells within the same multicellular organism. These variations are commonly called Variable Number of Tandem Repeats (VNTRs) (WRIGHT 1994).

The tandem repeats with the shortest repetitive units are called microsatellites. It is generally accepted that the length of the repeated motif in microsatellites ranges from 1 to 6 nucleotides, whereas motifs longer than 10 nucleotides are called minisatellites. This distinction is rather artificial (WRIGHT 1994) and thus varies among authors. The reason for this ambiguity is that, initially, microsatellites and minisatellites were explored mainly by researchers interested in using them as molecular markers, where the main difference among both was the technique used to analyze them. In general the total length of microsatellite alleles (80 to 500 nucleotides) could readily be analyzed by PCR (Polymerase Chain Reaction), whereas for the longer alleles of minisatellites (200 to >1000 nucleotides) analyses involving Southern hybridization were necessary. Furthermore, it is believed that the mutation mechanisms are essentially different among microsatellites and minisatellites, the first ones being affected mainly by replication slippage (LEVINSON and GUTMAN 1987b) and the later ones by recombination (JEFFREYS *et al.* 1988; JEFFREYS *et al.* 1994). However, an exact definition and distinction between microsatellites and minisatellites does not exist; in practice it is not clear where the upper limit for the replication slippage mechanism resides, and therefore there is a continuum between both entities in terms of genomic analyses.

1.3 Definition of microsatellites

Microsatellites are DNA sequence segments with special structural and mutational properties: they are formed by short nucleotide stretches repeated in tandem, and this repetitive structure makes them prone to errors during DNA replication (LEVINSON and GUTMAN 1987b). If the consequences of these replication errors are not corrected on time, they usually lead to increases or decreases in the number of tandem repetitions of the microsatellite involved. The specific mechanisms by which microsatellites mutate are described in section 1.6. Each repetitive unit of a microsatellite has a specific sequence of nucleotides; which hereforth will be referred to as "motif".

The term 'microsatellites' was coined by Litt and Luty (1989) who studied the abundance of $(TG)_n$ repeats as a smaller version of minisatellites. Earlier that decade, the special Z-DNA forming capacity and high polymorphism of microsatellites (then called "simple sequences") were shown by Hamada *et al.* (1982; 1984) and Levinson and Gutman (1987b), respectively. These last authors (LEVINSON and GUTMAN 1987b) also proposed that short tandem repeats,

in their case mono- and dinucleotides, mutate by a process called "slipped strand mispairing", and that the more repetitions a tandem repeat has, the higher will be its probability of mutating by slipped-strand mispairing.

Microsatellites occur across DNA sequences in every organism studied to date, and in higher numbers than expected by chance alone (FIELD and WILLS 1998; LAI and SUN 2003; TAUTZ *et al.* 1986). This extreme abundance is attributed to their repetitive structure, which was observed to pose more difficulties for the replication machinery than more random sequences. Microsatellite sequences were shown to cause delays or sudden stops in the polymerase reaction (HILE and ECKERT 2004), and to eventually impede the proper functioning of the polymerase complex by forming stable secondary structures among complementary repeats (COX and MIRKIN 1997). Tandem repeats in DNA were also shown to be favourable sites for DNA breakage and recombination (RICHARD and PAQUES 2000). Regardless of the processes involved in microsatellite mutations, multiple direct and indirect studies of microsatellite mutation have shown that mutations within microsatellite sequences, mainly gains or losses of complete repeat units, can be 5 to 8 orders of magnitude higher than other usual replication errors like nucleotide substitutions, insertions and deletions (ANMARKRUD *et al.* 2008; BROHEDE *et al.* 2002; BULUT *et al.* 2008; SEYFERT *et al.* 2008; THUILLET *et al.* 2002).

1.4 Microsatellite abundance and distribution within genomes

Microsatellites are abundant in all eukaryotic and prokaryotic genomes studied so far, but the validity of this statement is strongly dependent on the definition of microsatellite in terms of the minimum length threshold used to detect them throughout sequences. Microsatellites with less than 8 repeats are usually overrepresented in all genomic sequences studied to date (seeROSE and FALUSH 1998). However, when the minimum length threshold is increased (i.e. 10, 12, 14 repeats), prokaryote genomes show less overrepresentation of some microsatellite motifs, and longer repeats are scarce. For example, Gur-Arie *et al.* (2000) found that in the *Escherichia coli* genome mononucleotide tandem repeats seldom exceed 9 nt in length, and di-, tri- and tetranucleotide repeats rarely exceeded 12 nt in length. Nevertheless, microsatellites with 6 nt or more comprised 2.4% of the *E. coli* genome in the same study. Prokaryotic genomes are believed to be restricted to small sizes (<5 Mb) by strong selection for rapid replication; therefore repeats would be

removed unless they are favoured by selection (VAN BELKUM *et al.* 1998). However, as I discuss in section 1.7.1, there are numerous examples of microsatellites in prokaryotes favoured by selection to be long and therefore polymorphic, which serve as sources of functional diversity within coding regions.

In eukaryotic genomes, microsatellites are highly abundant, and this abundance decreases exponentially with the length of the repeat. Microsatellite abundance is not a function of genome size which, especially in higher eukaryotes, varies as a function of interspersed repeat content (TOTH *et al.* 2000). Conversely, a recent study by Takezaki and Nei (2009) presented evidence suggesting that total microsatellite variation can influence genome size in humans (approximately 10000 nt in size variation in chromosome 21). The total content of microsatellites within genomes does also not depend on the global CG composition of the sequence (KATTI *et al.* 2001; LIM *et al.* 2004). However, the global microsatellite content was shown to vary in a predictable way among species as measured by microarray signal intensities of microsatellite probes with more than 12 repeats, and this even correlates with established taxonomix relationships (GALINDO *et al.* 2009). Within a single genome, the total density of microsatellites is usually very similar among chromosomes, although sex chromosomes and smaller chromosomes in eukaryotes appear frequently as outliers, showing higher microsatellite content relative to the rest of the chromosomes.

The most conspicuous general pattern in microsatellite distribution in eukaryotes can be observed when binning microsatellites by motif length and nucleotide composition. Differences can be observed among genomic regions: introns, exons, intergenic regions depending mainly on the repetitive motif of the microsatellites. Microsatellites within exons tend to be more C/G rich, as coding regions in general have higher C+G content. Tri- and hexanucleotide microsatellites are overrepresented in coding regions in comparison to their representation in introns and intergenic regions (METZGAR *et al.* 2000; TOTH *et al.* 2000). Coding sequences need to be transcribed accurately to conserve the reading frame of the encoded proteins. Since the genetic code is made up of triplets, any nucleotide insertion or deletion which is not a multiple of three would disrupt the reading frame (METZGAR *et al.* 2000). Furthermore, not all combinations of triplet repeats are present in coding regions (LIM *et al.* 2004; MALPERTUY *et al.* 2003; SUBRAMANIAN *et al.* 2003; TOTH *et al.* 2000). The motifs of the most common repeats in mammals translate into aminoacids with a mixture of characteristics: polar (hydrophilic) aminoacids such as glutamine (most commonly encoded by CAG), serine (AGC), and glutamic acid (GAG); and non-polar aminoacids like proline

(CCG), leucine (CTG), glycine (GGC) and alanine (GCG) (KATTI *et al.* 2000). In particular, long tandem repeats of highly hydrophobic anino acids are not favored in proteins (KATTI *et al.* 2000). In humans, glutamine repeats are exceptional because they very often expand beyond 20 repeats, and in these cases they are frequently involved in the development of neurodegenerative diseases (more 100 repeats). In contrast, triplets with high content of tymine (T) are scarce or abscent; none of the 10 codons containing more than one T in the sense strand (the strand that gets transcribed) is reiterated. Finally triplet repeats with motifs ACT and ATC are usually absent because they translate into stop codons (SUBRAMANIAN *et al.* 2003). Therefore, microsatellite motif abundance within coding regions depends on codon usage and selective constraints on proteins, which can be different among species.

Introns and intergenic regions contain in general more mono, di and tetranucleotide microsatellites, with a predominance of A/T-rich motifs in most species examined to date (TOTH *et al.* 2000). In primates mononucleotide repeats are the most abundant, being twice as frequent as di and tetranucleotide repeats. In contrast, in rodents dinucleotides are the most abundant motif type, occurring about three times more frequently than mononucleotide repeats (TOTH *et al.* 2000). Dinucleotide repeats are the most abundant motif after mononucleotides and show the most conspicuous difference among life kingdoms: the AC motif predominates in mammalian genomes, in contrast to AT, which is the most abundant motif in plants (CASACUBERTA *et al.* 2000; MORGANTE *et al.* 2002). In fungi this relationship is more difuse, and high abundances of AT, AG and CG can be observed depending on the species, but in no case are AC repeats the most abundant ones (LIM *et al.* 2004; MALPERTUY *et al.* 2003). CG dinucleotides are extremely rare in eukaryotic genomes (KATTI *et al.* 2001)

Two regions within eukaryotic chromosomes are conformed almost exclusively by tandem repeats: centromeres and telomeres (KAPITONOV *et al.* 2004). Centromeres and subtelomeric regions are composed of repetitive sequences with longer motifs, and devoid of microsatellites; subtelomenric regions are almost completely covered by minisatellites (AMARGER *et al.* 1998; KAPITONOV *et al.* 2004). Telomeres, on the other hand, consist of microsatellite-like hexanucleotide tandem repeats associated with specialized proteins that occur at the ends of linear chromosomes. They serve as a protective mechanism to prevent chromosome shortening due to incomplete DNA replication, and to prevent these ends from recombining or from being recognized as double strand breaks and therefore getting degraded. The repeated motif in vertebrate telomeres is TTAGGG and the total length of these repeats varies among species and depends on cell type; ranging from 5 to 15 kilobase

(kb) in humans, and up to 200 kb in laboratory mice. These lengths are approximate since they cannot be determined precisely due to the difficulty in sequencing repetitive DNA.

A wealth of information on microsatellite abundance and distribution has been accumulated during the last 15 years. From this data it can be deduced that the whole-genome abundance of microsatellite types and motifs are unique and characteristic for the taxonomic group examined. Even closely related strains of *Cryptococcus neoformans* were observed to differ in motif abundance (LIM *et al.* 2004). Differences in repeat frequencies of various repeat classes can not be attributed to differences in nucleotide composition of the sequences, but structural properties (ability to form hairpin or quadruplex structures) of the microsatellite motifs were shown to influence their relative abundances (COX and MIRKIN 1997; GALINDO *et al.* 2009). Moreover, according to Galindo *et al.* (2009) particular microsatellite motifs are characteristic of one species versus another. All these observations suggest that differences in DNA replication and repair processes, and species-specific metabolic characteristics, among others (KARLIN *et al.* 2002) are responsible for the taxon-specificity of microsatellite abundance.

1.5 Microsatellite mutation mechanisms

The mechanisms involved in tandem repeat expansion and contraction, and the situations and conditions under which these mutations occur are not completely elucidated. Well before the scientific community developed a special interest for small tandem repeats, Fresco and Alberts (1960) showed that single strands from a double stranded ribonucleotide chain can form loops of more than one unpaired base due to mispairing with the complementary strand. They therefore proposed that the formation of this kind of loop could lead to insertions and deletions within the sequence. Levinson and Gutman (LEVINSON and GUTMAN 1987b) coined this process as Single-Strand Mispairing (SSM).

According to Levinson and Gutman (1987b) SSM occurs during replication when the two DNA strands are dissociated and the polymerase complex is copying each of the strands. (**figure 1.2 A**) Short tandem repetitions were shown to be more difficult to process for polymerases, at least in *in-vitro* DNA synthesis (HILE and ECKERT 2004). This is due to pausing of the polymerase during copying of the tandem repeat, which can eventually lead to a dissociation of the polymerase complex. The unpaired single-stranded DNA at the dissociation point can undergo re-annealing 'out of frame' because the repetitive units of the microsatellite can anneal at different places within the tandem repeat. If the loop and

re-annealing occurs in the nascent strand, it can produce an expansion, and if it occurs in the template strand, the results will be a contraction. Such loops can be repaired by the well-characterized mismatch repair (MMR) system and it has been shown that MMR deficiency is associated with microsatellite instability in a variety of organisms (HARR *et al.* 2002; LEVINSON and GUTMAN 1987a; SIA *et al.* 1997; STRAND *et al.* 1993). Additional support for a role for replication errors in microsatellite mutations is seen in *in vitro* replication of DNA strands containing microsatellites (BAKKER 2005; SHINDE *et al.* 2003). This often results in a proportion of replicated tracts with higher and lower repeated copy numbers than the original.

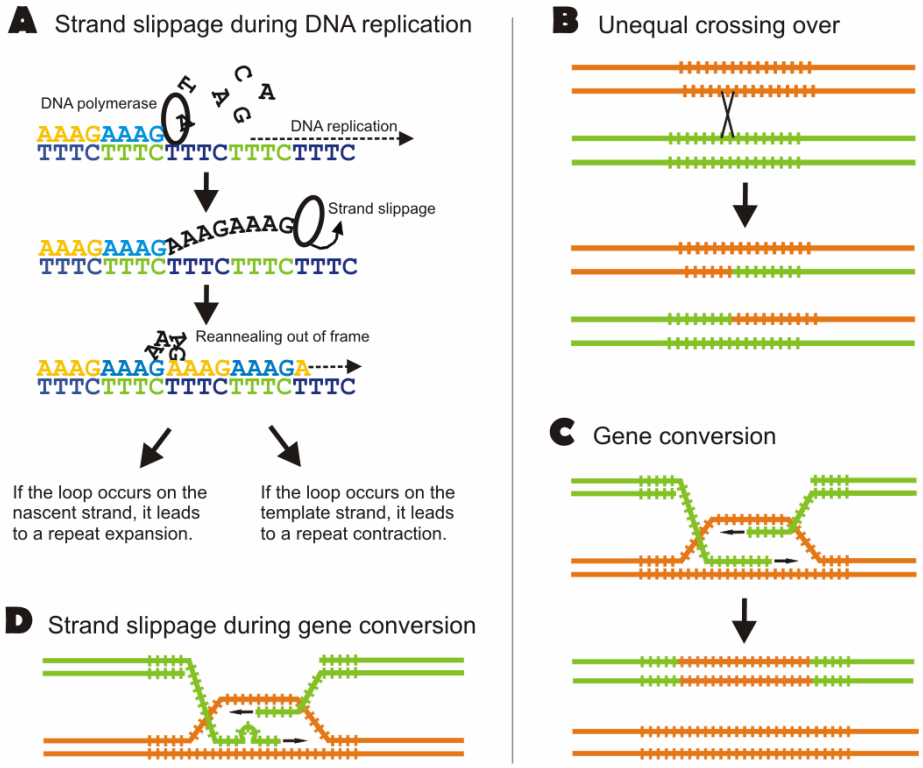

Figure 1.2: Depiction of mechanisms believed to be involved in microsatellite hypermutations. A: strand slippage during replication or single strand mispairing, B: recombination with unequal crossing over, C: recombination with gene conversion, D: strand slippage during recombination.

A second process initially proposed to be involved in microsatellite hypermutation is recombination between DNA strands by unequal crossing over (SMITH 1976). This process was expected to produce changes in microsatellite length in the same way it did in the case of minisatellites (RICHARD and PAQUES 2000) (**figure 1.2 B**). However, it has been shown that it is mostly gene conversion that produces changes in microsatellite tandem repeat numbers (RICHARD *et al.* 2000; WELCH *et al.* 1990). Gene conversion involves an unidirectional

transfer of information by recombination from one DNA helix (which remains unchanged) to a second DNA sequence (which gains a fragment copied from the first one), as shown in **figure 1.2 C**. The effect of gene conversion seems to be most relevant in trinucleotide repeats, especially in large trinucleotide expansions that are involved in neurological diseases like myotonic dystrophy and Fragile X (JAKUPCIAK and WELLS 1999; JAKUPCIAK and WELLS 2000a; JAKUPCIAK and WELLS 2000b).

Figure 1.2 illustrates how the two mentioned mechanisms would generate changes in the number of tandem repeats of a microsatellite. These two mechanisms are not mutually exclusive, but SSM is believed to be responsible for the majority of microsatellite mutations. based on empirical (HUANG *et al.* 2002; KAYSER *et al.* 2000; KELKAR *et al.* 2008; SUNDSTROM *et al.* 2003) and simmulation data (DIERINGER and SCHLÖTTERER 2003; ZHU *et al.* 2000a).

The question of whether heterogeneous mutational mechanisms are a significant factor underpinning the differences in mutation rates and patterns between microsatellite loci remains open to some degree. Most of the studies looking closely at mutational mechanisms have been of those microsatellites involved in heritable human disease or cancer, and these may represent a biased sample. Further experimental studies are necessary to quantify the relative influence of recombination and slipped strand mispairing on microsatellite mutation. The distinction is complicated by links between the two processes, since strand misalignment can occur during the stages of recombination that involve strand exchange between chromosomes and subsequent synthesis of DNA (**figure 1.2 D**).

1.5.1 Factors affecting microsatellite mutation rates

Microsatellite mutations can occur during chromosomal replication, either as part of mitotic or meiotic processes, or during repair or recombination processes that require DNA synthesis. Therefore the frequency of microsatellite mutations increases in rapidly dividing cells or under stress conditions when cells undergo active repair due to damage. Any mutation arising during meiosis or in the initial mitotic divisions of an embryo will proliferate and, if not selected against during development, will constitute a new microsatellite allele. In contrast, mutations arising in differentiated cells constitute somatic mutations which will not affect other cells unless the affected cell starts dividing again, as is the case in cancerous cells. To the current knowledge, there is a range of factors interacting

to affect microsatellite mutation rate during these processes, and these have been the focus of numerous studies, but only a few of them have effects prone to generalization.

Microsatellite mutations are basically errors which occur during cellular processes involving nucleic acid polymerization (DNA replication, gene conversion, some kinds of DNA repair). The vast majority of newly arisen mutations are quickly corrected by DNA repair processes (EISEN 1999). However, microsatellites usually have mutation rates substantially higher than other parts of DNA sequences (~10^{-3} mutations/locus/generation in microsatellites vs 10^{-9} mutations/locus/generation in SNPs), and this would indicate that the process of microsatellite mutation poses a higher degree of difficulty for repair processes than other kinds of mutations.

Most microsatellite mutations arising by replication slippage are quickly corrected, mainly by the MMR system, rendering MMR efficiency as one of the main factors affecting microsatellite mutation rate (STRAUSS 1999). However, the efficiency of repair declines as the size and/or stability of the loop formed by the repeats during replication slippage increase. These two characteristics, in turn, are affected by other factors intrinsic to the microsatellite: the allele length, motif length, nucleotide composition, and imperfections within the microsatellite. Moreover, the relative importance of these factors is very likely to vary depending on the genomic position of the repeat and probably even by lifestyle characteristics of the individual (TYSON and MATHERS 2007). In sexually reproducing organisms, especially those with a relatively long life span, the gender and age of the individual can also affect the generation of microsatellite mutations, and the transmission of these mutations to the offspring (e.g. BECK *et al.* 2003; e.g. MAKOVA and LI 2002). The probability of recombination events within microsatellites can also be affected to different extents by the mentioned factors. For simplicity, the most important factors affecting microsatellite mutation are considered individually below. However, it should be kept in mind that the interactions among these factors are dynamic and take place at different levels simultaneously, as depicted in **figure 1.3**.

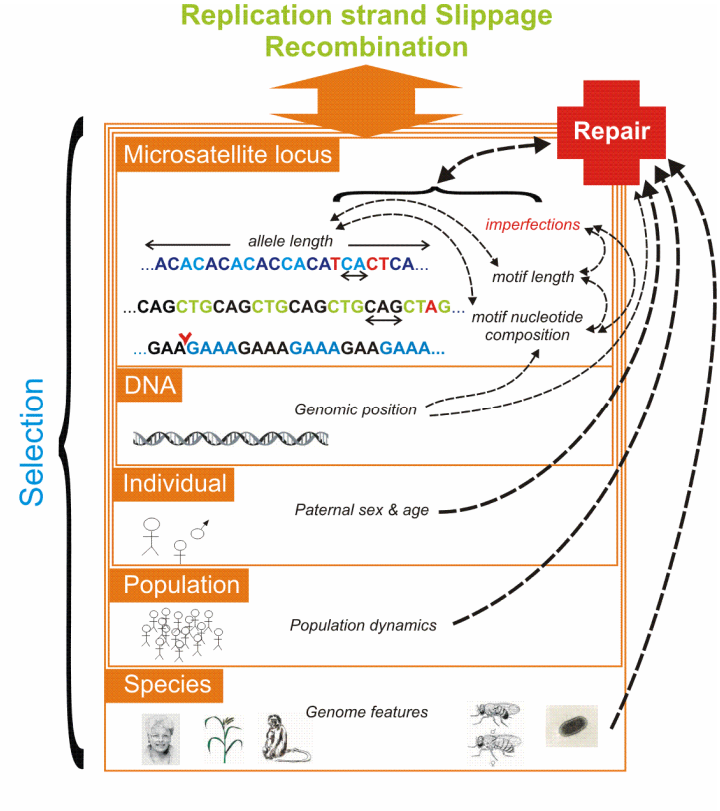

Figure 1.3: Factors and processes affecting microsatellite mutation. Factors (*italics*) operate at different hierarchical levels (orange boxes), starting from the smallest scale the microsatellite locus itself and moving up to the species level. Selection operates across all levels. All these factors interact dynamically, affecting the rate of replication slippage and recombination, the DNA repair processes and, therefore, microsatellite variability.

DNA repair

DNA repair is essential for the maintainance of DNA integrity, preventing mutations which could compromise the adequate functioning of the organism. Failure of the MMR system during replication can result in up to 10^3 fold increase in microsatellite instability (STRAND *et al.* 1993). Microsatellite instability was the first clue indicating a failure in MMR in certain types of tumors, like colorectal cancer (UMAR *et al.* 1994). Defects in the exonucleolytic proofreading activity of DNA polymerases have less impact on microsatellite mutation, with a 5 to 10 fold increase in mutation rate (STRAND *et al.* 1993; STRAUSS *et al.* 1997). Further, repair activity is not uniform throughout the genome, and MMR efficiency has been found to be strand and substrate specific. The resulting instability will depend on the repair specifity of the MMR proteins and on the microsatellite motif involved.

Furthermore, loss of specific families of MMR genes will lead to different mutational biases towards expansions or contractions (SHAH *et al.* 2010).

Not only does the MMR system afect microsatellite mutation rates in a variety of ways, but microsatellite mutation can also affect the MMR system. In eukaryotes, several of the minor MMR genes contain mononucleotide microsatellites of 7 to 9 repeats within their coding regions, which constitutes a rare and rather paradoxic occurrence (CHANG *et al.* 2001). Expansions or contractions within these genes can deactivate them, rendering a less efficient MMR protein complex. Under normal circumstances, this would provoke a reduction in fitness owing to increased deleterious mutations. However, as is the case in contingency loci from bacterial pathogens (BAYLISS *et al.* 2004), under circumstances favoring rapid evolution, an increase in mutation rate would also increase the rate of necessary adaptive mutations. Therefore, Chang *et al.* (2001) hypothesize that the exceptional density of microsatellites within MMR genes represents a genetic switch that allows the adaptive mutation rate to be modulated over evolutionary time.

Microsatellite length

The most accepted factor influencing microsatellite mutation is the overall length of the repeat array. Longer alleles tend to have higher mutation rates than shorter ones, showing a positive exponential relationship between the number of repeat units and the microsatellite mutation rate (FALUSH and IWASA 1999; KELKAR *et al.* 2008; SIA *et al.* 1997; WIERDL *et al.* 1997). This pattern was observed initially in experimental bacterial systems (STREISINGER and OWEN 1985), and it was subsequently coined "dynamic mutation" because the probability of mutation will change with every mutation. The likely explanation for this phenomenon is that longer repeat arrays offer more opportunities for misalignment during replication or recombination, and can also produce longer and more stable loops. For example in trinucleotides, the conformational entropy of slippage is ~2 kcal/mol more destabilizing for longer repeat arrays (HARVEY 1997)

Additionally, the direction of mutation, contraction or expansion, was observed often to vary with array length. In the case of mutations resulting from SSM, the direction of mutation (contraction or expansion) will depend on the DNA strand where the outstanding loop forms, i.e. short microsatellites tend to experience more expansions than contractions, whereas longer microsatellites are occasionally subject to large deletions (XU *et al.* 2000) The first case, where microsatellites tend to expand, could be explained by a bias of loop formation in the leading strand during DNA replication. Large deletions, on the other hand,

are more likely to occur by recombination; the longer the microsatellite, the higher the probability of non-homologous alignment during meiotic recombination. However, the "directionality" of microsatellite mutation is not yet clear, and could be produced by the interaction of several other factors.

Motif length

Shorter repeat units can, in theory, allow more possible slippage events per unit length of DNA, and are therefore expected to have higher mutation rates. Motifs more than three nucleotides long require higher dissociation energy, and are thus less likely to generate enough single-stranded DNA to form a stable loop (HARVEY 1997). However, motif length can affect MMR efficiency. If the loop is too big (i.e. more than 18 bp) the efficiency of MMR drops (JENSEN *et al.* 2005). Here, the effect of motif length becomes evident, since longer motifs form bigger loops.

Both situations, longer motifs being more mutable (ANDERSON *et al.* 2000 ; WEBER and WONG 1993), and shorter motifs being more mutable (JOUQUAND *et al.* 2000; SIA *et al.* 1997) have been observed, and there exists a lot of variation with respect to this factor. A recent study by Kelkar *et al.* (2008) reported that, total microsatellite length being equal, mutability is conspicuously lower for microsatellites with longer motifs. Among microsatellites with high repeat numbers (n > 15), the mutability is lowest for mononucleotides, followed by dinucleotides, and highest for tri- and tetranucleotides Therefore, the influence of motif length may be overshadowed by other factors.

Motif nucleotide composition

The thermostability and conformational properties of a DNA sequence depend strongly on its nucleotide composition (SHAH *et al.* 2010). Kelkar *et al.* (2008) showed recently that, among mononucleotides, the mutability of poly-A/T repeats is significantl higher than for poly-C/G repeats, at least for repeat numbers lower than 17. In experiments with cultivated cell lines, $(C)_8$ repeats showed to be more unstable than $(A)_8$ repeats in functional functional MMR systems (SAGHER *et al.* 1999). Mutation rates increase for all these repeats when the MMR system is not functional, but these increases differ depending on the nucleotide composition: mutation rate for poly-C/G and poly-CA/GT repeats was found to be about 7-15 fold higher than that for poly-A/T repeats in MMR deficient cells (BOYER *et al.* 2002).

Repeats with certain motifs have a heightened propensity to form secondary structures or alter DNA structure. Secondary structures, such as hairpins, quadruplex structures, H-DNA or sticky DNA, being intermediate DNA hybrid forms, increase the likelihood of strand misalignment and subsequent polymerase slippage. For example, perfect or near perfect homopurine/homopyrimidine mirror repeats can form H-DNA, and alternating purine-pyrimidine tandem repeats form Z-DNA (COX and MIRKIN 1997). Conformational changes in these repeats can occur during DNA replication when the complementary DNA strands dissociate, hindering in this way the action of DNA polymerases and repair enzymes. Some sequences, like the spinocerebellar ataxia-causing (ATTCT)·(AGAAT) element, even have the potential to unwind DNA locally, promoting single-stranded DNA which highly facilitates secondary structure formation (LIN and ASHIZAWA 2005). In this context, a study by Bacolla *et al.* (2008) based on both, empirical and genome sequence data, showed that the relative abundance of tri-and tetranucleotide repeats are inversely proportional to the capacity of their single strands to fold back into hairpin or quadruplex structures of varying thermodynamic stabilities.

Imperfection and interruptions within the microsatellite

Point mutations and other interruptions within the repeat array have been observed to reduce mutation rate, which is most likely due to an overall reduced chance of slippage, secondary structure formation, and/or recombination, when imperfections are interrupt tandem arrays (DETTMAN and TAYLOR 2004; SAKAMOTO *et al.* 2001; SYMONDS and LLOYD 2003; VAN TREUREN *et al.* 1997)}. However, repeat arrays with several motifs, known as compound or complex/clustered microsatellites, show elevated mutation rates for at least one of the internal motifs, when compared to a microsatellite of the same length and motif. One explanation for the phenomenon is that both fractions, although containing different motifs, have similar structural propensities (BULL *et al.* 1999).

Genomic context

The mutability of any DNA sequence depends on its context within a genomic sequence. This is most apparent when observing the distribution of microsatellites in coding regions where the effect of mutations has a high probability of being disadvantageous and is therefore strongly counteracted by selection (METZGAR *et al.* 2000). Alternatively, mutation rate variation can arise through structural propensities of either flanking sequences or even more distantly neighbouring regions, being most likely based on the thermodynamic propensities of different base-pairings ($\Delta T_m GC > \Delta T_m AT$). A few studies have shown that the

propensity of expansion of certain types of microsatellites, namely GC-rich trinucleotide repeats, is positively correlated with GC-bias of the flanking sequence (TEMNYKH *et al.* 2001), but others found no evidence for such a correlation. Further, CpG islands (CG dinucleotides) are found in many mammalian promoters and are, when methylated, involved in chromatin remodelling and gene silencing. The observed proximity of some highly expandable loci to CpG islands has led to the suggestion of a mechanistic link between these elements and microsatellite instability (AHUJA *et al.* 1997; GARGANO *et al.* 2007; KIM *et al.* 2005).

Sex and age

Sex has been observed to affect overall microsatellite mutation rate, and this might be driven by processes that are specific to sperm or oocyte development. Both, sex and age, have been observed to affect the probability of transmission of a mutated allele (ELLEGREN *et al.* 1995). For example, human males produce considerably more gametes than females and show therefore more cumulative germline cell divisions at an older age than at a younger age (HURST and ELLEGREN 1998; KAYSER *et al.* 2000). In contrast, the female reproductive system stops producing ovules after birth, therefore being exposed to fewer mutations associated with DNA replication, and having no significant age effect. Supporting studies have found that male reproductive cells mutate five times more often than female ones, and older men pass on more mutations than younger men. Studies in other species, for example fish, show fewer differences among male and female transmitted mutations, because the ratio of male to female gametes is smaller (NEFF and GROSS 2001).

1.6 Origin of microsatellites

One of the main hypotheses proposed to explain microsatellite genesis regards the fortuitous generation of repeated motifs within random sequences by point mutations or small insertions and deletions (SCHLÖTTERER 2000; ZHU *et al.* 2000b). Once a "proto-microsatellite", with two or three repeats, has arisen, its maintenance and growth is expected to be favoured by its propensity to undergo strand slippage during DNA replication and repair processes (LEVINSON and GUTMAN 1987b) and, depending primarily on the nucleotide composition and length of the repeated motif, its capacity to form unusual DNA conformations (COX and MIRKIN 1997) and to participate in recombination and transposition events (JAKUPCIAK and WELLS 2000b; RICHARD *et al.* 2000; RICHARD and PAQUES 2000). As discussed above, the number of repeat units correlates positively with the mutability of the microsatellite, but the minimum repeat number necessary to allow for

strand slippage or other mechanisms involved in microsatellite mutation to occur is debatable. Initially, eight repeats were suggested as the minimum threshold for a small tandem repeat to be considered a microsatellite (ROSE and FALUSH 1998), and therefore smaller microsatellites were left out of most studies in eukaryotes. In prokaryotes, however, the majority of short tandem repeats has less than eight repeats, and microsatellites with as few as two repeats were shown to be polymorphic in *Mycobacterium* species (SREENU *et al.* 2006).

A second widely accepted hypothesis regards the dispersion of sites for microsatellite origin by transposable elements (especially retrotransposons). Transposable elements are sequences that have the capacity to "jump" (transpose) to different positions in the genome generating multiple copies of themselves. These can be divided into two main classes based on their mechanisms of movement. Class I are retrovirus-like transposons that get transcribed into messenger RNA and subsequently retro-transcribed back to DNA and inserted in a new position in the genome. Class II are so called "cut and paste" transposons because they get excised from their original position and inserted into a new position. Both of these elements can leave traces of their presence and movement during the transposition process across DNA sequences, which resemble microsatellites (e.g. contain small tandem repeats), especially poly-A arrays. Class I retrotransposons get a poly-A tail added at the 3′ end after transcription into mRNA, which then gets inserted together with the transposed sequence into the new position. Retrotransposons can also contain other microsatellite-like stretches within their sequences including dinucleotide and tetranucleotide repeats. Class II transposons insert preferentially into certain DNA sequences which can be either inverted repeats or tandem repeat sequences. This suggests a reciprocal association in which microsatellites act as "retroposition navigator sequences" while retrotransposons generate more microsatellites during their dispersion through the genome (NADIR *et al.* 1996).

A good example of retrotransposon mediated microsatellite genesis in humans is the well documented origin of A/T-rich microsatellites with motifs ranging from one to six nucleotides in length from *Alu* elements (ARCOT *et al.* 1995; BATZER *et al.* 1995; NADIR *et al.* 1996). *Alu* repeats are the most abundant interspersed repetitive elements in primate genomes, and are comprised of two monomers separated by a poly-A tract. These retrotransposons also have the typical poly-A tail at their 3′ end. Both of these repeats give rise to poly-A and A-rich microsatellites (i.e. AAC, AAG, AAAAT), and dinucleotide microsatellites (i.e. AT, AC, AG). The 3′ end poly-A tail tends to be longer than the middle one in humans, giving rise to the mayor part of microsatellites arisen from *Alu* elements.

The association of poly-A and A/T-rich microsatellites with transposable elements may, at least partly, explain the fact that A/T rich motifs are by far the most abundant repetitive arrays within genomes (see KATTI *et al.* 2001; SUBRAMANIAN *et al.* 2003; TOTH *et al.* 2000). On the other hand, GC-rich microsatellites, especially trinucleotide and hexanucleotides do not seem to be associated with transposable elements (NADIR *et al.* 1996). Rather it was suggested that the origin of trinucleotide repeats could be associated with the process of codon reiteration in the evolution of proteins, a process which favours increases in protein size by expansion of repetitive domains (GREEN and WANG 1994; MAR ALBÀ *et al.* 1999).

As in the case of mutation mechanisms, the origin of microsatellites is a topic which remains open for new theories. Probably the first to refer to the origin of microsatellites was Tautz (1986) with the words "microsatellites are born from regions of 'cryptic simplicity', i.e. regions in which variants of simple repetitive DNA sequence motifs are already represented". However, the opposite scenario, where regions of cryptic simplicity are generated during the process of expansion and degradation of microsatellite regions, is also likely.

1.7 Phenotypic effects of microsatellite mutations

For a great part of the last half a century the functional genome was regarded as those sequences coding for proteins and describing high conservation (i.e. high sequence similarity) among taxa. Thus functional analyses were focused on coding regions that, at least in human, mouse, and chimp, account for less than 2% of the genome. The fact that coding exons usually lack microsatellites, or contain mostly trinucleotide repeats, led to the idea that microsatellite variation is either deleterious or restricted to non-functional intergenic DNA.

As is usual with genetic studies, the first outstanding findings about phenotypic effects of microsatellites were related to human health. Early in the 19th century, two neurodegenerative diseases, namely fragile X syndrome and X-linked spinal and bulbar atrophy, were associated with extreme expansions detected within trinucleotide repeats located within gene coding sequences. Since then, nineteen congenital neurological diseases have been found to be associated with the instability of microsatellites (BROUWER *et al.* 2009). Most of these repeats are trinucleotides, but there are also tetra- and pentanucleotides involved. These microsatellites are situated within genes, both within translated and untranslated regions, and the pathological effects are due to spontaneous

and excessive expansion of the repeats, causing gain or loss of function. **Table 1.1** shows a summary of these diseases and the repeat motifs involved. For a more detailed description of the genetic consequences of microsatellite expansions involved in neurologic diseases see the review by Brouwer *et al.* (2009), and for clinicopathologic information on clinical symptoms of the diseases have a look at Orr and Zoghbi (2007).

Table 1.1: Human genetic diseases associated with overexpansion of microsatellites, based on (CUMMINGS and ZOGHBI 2000; EVERETT and WOOD 2004; LIN and ASHIZAWA 2005; RICHARD *et al.* 2008)

Disease	**Repeat motif**	**Normal**	**Disease**	**Genomic Region**
Fragile X	$(CCG)_n$	6-53	>230	5′-UTR
FRAXE	$(CCG)_n$	6-35	>200	5′-UTR
Spinal and bulbar muscular atrophy (Kennedy's disease)	$(CAG)_n$	9-36	38-62	exon
Huntington`s disease	$(CAG)_n$	6-35	36-121	
Machado-Joseph disease	$(CAG)_n$			
Dentatorubral-pallidoluysian atrophy	$(CAG)_n$	6-35	49-88	
Myotonic dystrophy 1	$(CTG)_n$	5-37	>50	3′-UTR
Myotonic dystrophy 2	$(CCTG)_n$			intron
Spinocerebellar ataxia 1-7	$(CAG)_n$			exon
Spinocerebellar ataxia 8	$(CTG)_n$	16-37	110 to <250	3′-UTR
Spinocerebellar ataxia 10	$(ATTCT)_n$	10-29	800-4500	intron
Spinocerebellar ataxia 12 (LIN and ASHIZAWA 2005)	$(CAG)_n$	7-28	66-78	5′-UTR
Progressive myoclonous epilepsy	$(C_4GC_4GCG)_n$			
Friedreich`s ataxia	$(GAA)_n$	7-34	>100	intron
various human cancers	minisatellite, di-,tri- & tetra-nucleotides			
Hereditary Non-polyposis Colon Cancer	*mono-,di- and tri-nucleotides*			

The adaptive value of microsatellite polymorphism was first explored in detail in prokaryotes, especially pathogenic bacteria (MOXON *et al.* 1994). Genetic variation in pathogenic bacteria is generally restricted because infections often result from propagation of a few founder cells. Low genetic and phenotypic variability would render the pathogens highly vulnerable to changeable immune responses in the host. However, many pathogenic bacteria can still generate variation in certain genes thanks to the presence of microsatellites within their coding or regulatory regions. Since microsatellites mutate frequently and reversibly, they can inactivate the functional domain of a gene (i.e. by

frameshift mutation) in one generation and mutate back to reactivate it in the next one. Genetic loci affected by this kind of 'on/off switching' from microsatellites are called contingency loci (BAYLISS *et al.* 2004). Many of these genes code for surface molecules, called "virulence factor genes", which have an essential role in the cell's interactions with its environment, and therefore a very high impact on microbial fitness. Examples of virulence factor genes are genes encoding for proteins in the capsule which confer serum resistance, pili proteins which affect cell adhesion, and other surface proteins affecting the formation of surface pores and nutrient acquisition. In *Haemophilus influenzae* and *Staphylococcus aureus*, changes in the length of microsatellites within virulence factor genes result in conformational changes in processed surface proteins, which make these unrecognizable to the host's antibodies (MOXON *et al.* 2006). A similar situation can be observed in eukaryotes such as the yeast *Saccharomyces cerevisiae*, in which more than 75% of genes containing microsatellites encode cell wall proteins. Variation in repeats associated with these genes gives rise to quantitative alterations in phenotypes (e.g. adhesion, flocculation or biofilm formation) (VERSTREPEN *et al.* 2005).

In populations of single-celled species, mutations generating variability can immediately favour adaptation by increasing the chance that at least a few individuals will be able to survive under stress conditions. If a microsatellite mutation is deleterious, the death of a single cell will not endanger the rest of the population. However, multicellular organisms should be more intolerant to mutations in microsatellites or elsewhere because each cell is part of a complex system. This problem was partly overcome by increasing proteome diversity by segmental duplications, diploidy and polyploidy, and the implementation of alternative splicing, which requires a concomitant increase in regulatory information. In contrast to single celled organisms, multicellular eukaryotes have extensive intronic and intergenic sequences whose extent increases with developmental complexity (TAFT *et al.* 2007). Interestingly, the majority of the genome (including non coding regions) gets transcribed during some stage in development, but most of these transcripts, including a great proportion of transcripts from coding regions, are not translated into proteins. Instead, they constitute introns, 5' and 3' UTRs (untranslated regions), or remain as RNA regulating cell functions (WONG *et al.* 2001).

Due to the intricate ways in which DNA sequences and protein complexes interact to fulfill cellular functions, the repetitive structure and frequent length variation of microsatellite sequences, both within and outside coding regions, can influence chromosome structure, gene expression, protein function and even DNA repair and

recombination in multiple ways which are broadly outlined in **figure 1.4**, and described below.

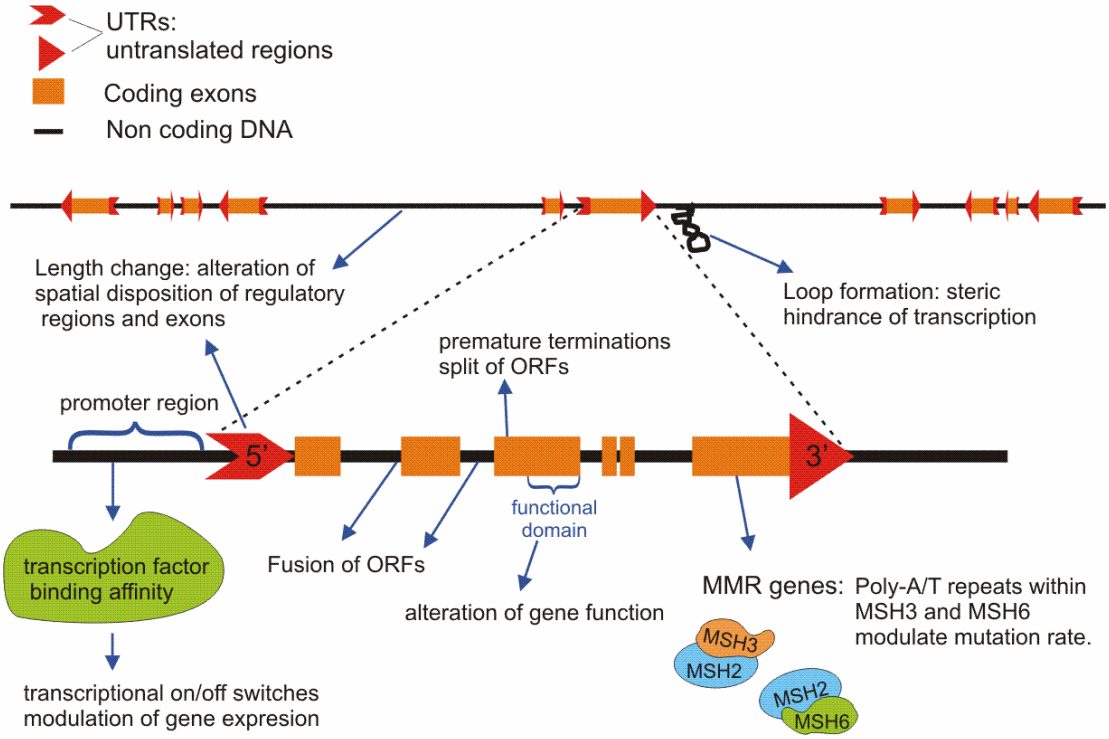

Figure 1.4: Functional implications of microsatellite length change. Microsatellite length variations have been shown to mediate diverse functions depending on the genomic region in which these are present. Within exons microsatellite mutations can induce changes in protein structure, therefore altering its function, or can directly inactivate the protein by trunctation or fusion of open reading frames (ORFs). Within introns and intergenic regions these changes can partake in the modulation of gene expression, either by modifying the structure of transcription factors or enzymes involved in transcription modulation, or by changing the secondary and/or tertiary structure of DNA or RNA regions that interact with transcription factors. Furthermore, microsatellites are involved in the regulation of their own and genome wide mutation rates, by being present within the minor components of the mismatch repair system.

1.7.1 Effects of microsatellites within exons

Coding repeats, especially microsatellites, are targeted by numerous studies because their instability has been shown to lead to genetic diseases such as Huntington disease and Friedreich's ataxia. However, the relatively high incidence of trinucleotide microsatellites within exons suggests that these are not being eliminated by selection due to potential benefit for the cell. Indeed, microsatellites are markedly overrepresented in transcription factors, protein kinases and genes encoding developmental regulatory proteins. Fondon and Garner (2004) hypothesized that the

impressive range of phenotypic variation observed among dog breeds was due to length variations of microsatellites within developmental genes. As part of their study they found that *Runx-2,* a gene coding for a transcription factor which, in vertebrates, regulates the differentiation of osteoblasts, has two homopolymeric tracts side by side within its aminoacid sequence: polyglutamine (18-20 repeats) and polyalanine (12-17 repeats). Coded by two perfect trinucleotide repeats, the ratio of repeat lengths of these alleles correlated strongly with the downward bending of the dog's muzzle in several dog breeds. Another gene, the *Alx-4* gene, contains an imperfect hexanucleotide repeat coding for poly-proline-glycine stretch. A 51 bp deletion in this repeat destroys the binding ability of the *Alx-4* protein to bind to the lymphoid enhancer binding factor-1 to target gene expression in limb bud mesenchyme (FONDON and GARNER 2004). The consequence in both mice and dogs is the development of an additional digit in the hind feet (polydactyly). Like these two examples, an increasing number of studies are uncovering the effects, either positive or negative, of repeat expansions within exonic regions. Polyglutamine peptides have been shown to drive transcription while polyalanines repress transcription in a length dependent fashion. The processes affected are generally regulatory, influencing directly or indirectly the expression of proteins which act at different levels of enzymatic cascades.

Another example, where the effect of microsatellite mutation within genes has genome-wide effects, is the case of microsatellites within the mismatch repair system genes. These encode an enzymatic complex which is highly conserved with close homologues between eukaryotes and prokaryotes, and is involved in the correction of base pair mismatches and mutations due to strand slippage and loop formation during replication. The coding regions of the minor MMR genes (hMSH3 and hMSH6 in humans) contain several mononucleotide repeats (mainly poly-A/T), and variations in the length of these stretches permit the modulation of mutation rates over evolutionary time (CHANG *et al.* 2001). The MMR system is normally extremely efficient and, therefore, microsatellite length in somatic cells tends to be stable. However, if the MMR system becomes defective or overwhelmed due to external factors (e.g. mutagenic agents), cells start accumulating altered microsatellites by thousands, a phenomena known as microsatellite instability, which is involved in cancer development.

1.7.2 Effects of microsatellites in introns and non-coding regions

Non coding DNA might contain the majority of regulatory DNA. The concept of regulatory region is not yet well defined; a promoter region, for example, is a site in DNA where RNA polymerase binds to start transcription. The promoter could be several kb away from the transcription start site and is generally difficult to recognize based on DNA sequence only. Furthermore, regulatory sequences seem to have an elevated turnover rate; transcription factor-DNA interactions are highly polymorphic, and regulatory interactions are constantly gained and lost within populations. On average, humans are heterozygous at more functional cis-regulatory sites (>16,000) than at amino acid positions (<13,000), in part because of an overrepresentation among the former in multiallelic tandem repeat variation, especially AC dinucleotide microsatellites. The role of microsatellites in gene expression variation may provide a larger store of heritable phenotypic variation, and a more rapid mutational input of such variation than has been realized (ROCKMAN and WRAY 2002) .

An interesting example is the involvement of a complex microsatellite (e.g. $(CAGA)_n$, $(CATA)_n$, $(AG)_n$ and $(GAGGAGA)_n$ interspersed among non repetitive sequences) in the modulation of social behaviour. The microsatellite is immersed in the 5' regulatory region of the vasopressin 1a receptor (V1aR), which mediates the expression of the hormone vasopressin. Among other functions, vasopressin is implicated in memory formation and social behaviour in vertebrate species. Varying degrees of social interaction in voles (genus *Microtus*) were found to correlate with differing levels of vasopressin receptor expression in the brains of these species, and this in turn, with the size of the microsatellite. (HAMMOCK and YOUNG 2005). Praire and pine voles have a long version of the microsatellite (430 bp in total), and show high levels of social interest (i.e. the males are monogamous). In contrast, montane and meadow voles, which possess a truncated version of the microsatellite, are socially indifferent and the males do not contribute to parental care. Further, the capacity of the microsatellite to drive V1aR expression was demonstrated by *in vitro* luciferase reporter assays. In humans, four polymorphic microsatellites surround the human vasopressin reporter homologue, which suggests that behavioural variation in humans is likely to be subject to complex and highly variable regulatory interactions.

Microsatellites can also affect the structural properties of DNA. Expansions or contractions of a microsatellite change the length of the DNA sequence and consequently the spatial disposition of transcription factor binding sites with respect to exons and other transcription factors. Furthermore, the structure-forming potential of tandem repeats has the capacity to generate steric effects, favouring or disfavouring the access of transcription enzymes to particular coding regions.

One of the key applications of microsatellites as molecular markers is the construction of linkage maps for gene and QTL mapping. These applications are rooted on the major assumption of microsatellite neutrality, and microsatellite variation is used to identify linked genomic regions possibly involved in the generation of quantitative phenotypic variation. Recent evidence on microsatellite functionality, especially the potential of microsatellites to be involved in multiple processes of gene regulation, suggests the possibility that those microsatellites associated with QTLs are the actual effectors of the phenotypic variability observed in QTL analyses.

Evolution is a trade-off between gaining diversity in function and escaping the deleterious effects of mutations. Natural selection will favour the "fittest" individuals within a population, but which individuals are the fittest can be redefined suddenly depending on environmental influences. Because environmental changes occur stochastically and are unpredictable, fitness is dependent upon the available diversity in any limiting characteristic during situations of stress. In these situations, high mutability can be useful for the generation of genetic diversity. However, accumulation of random mutations where most of these are likely to be deleterious is likely to reduce fitness. Microsatellite mutations affecting protein function or expression can be regarded as "strategic mutations" because, besides of occurring at higher rates, these length mutations are gradual and fully reversible, and are ubiquitously available, therefore enabling rapid evolutionary adaptation.

The majority of microsatellites across a genome might not have a defined or critical role, because microsatellite sequences are likely to arise and expand at higher rates than their recruitment for functionality. However, variation in microsatellites is generated constantly and constitutes a rich reservoir of genetic variation. It is the intrinsic variation within these sequences, both in functional and non-functional regions, what underlies the evolutionary importance of microsatellites.

1.8 Conclusion

Microsatellites were once considered random casualties across DNA sequences playing no important role in genome functioning. In this sense, it was clear that microsatellite variation offered a baseline of neutral and constant evolution upon which species could be compared directly or indirectly in phylogenetic trees. However, as is usual with the advance of biological sciences, the more we find out about a research topic, the less we seem to know about it. Currently, microsatellite evolution is referred to as a rather complex process, and the models developed for its analysis are not applicable in most cases. Despite intense exploration of microsatellite mutation mechanics, the processes governing these changes and their effects on the rest of the genome are still poorly understood. We know that certain microsatellite loci can be involved in the onset of neurodegenerative diseases, while other loci can influence DNA transcription and translation processes, as well as protein conformation in the case of microsatellites within coding regions. However, it si difficult to draw conclusions without an established framework on microsatellite abundance, distribution, and classification. The availability of complete genomic sequences for an increasing number of eukaryotic species, and for an exponentially growing number of prokaryotic genomes, will make it possible to perform comparative microsatellite analyses between and within genomes, to further our knowledge about microsatellite evolution.

1.9 References

AGAPITO, J., J. RODRIGUEZ, P. HERRERA-VELIT, O. TIMOTEO, P. ROJAS *et al.*, 2008 Parentage testing in alpacas (*Vicugna pacos*) using semi-automated fluorescent multiplex PCRs with 10 microsatellite markers. Anim Genet **39:** 201-203.

AHUJA, N., A. L. MOHAN, Q. LI, J. M. STOLKER, J. G. HERMAN *et al.*, 1997 Association between CpG island methylation and microsatellite instability in colorectal cancer. Cancer Res **57:** 3370-3374.

AL-JAWABREH, A., S. DIEZMANN, M. MULLER, T. WIRTH, L. F. SCHNUR *et al.*, 2008 Identification of geographically distributed sub-populations of *Leishmania* (Leishmania) major by microsatellite analysis. BMC Evol Biol **8:** 183.

AMARGER, V., D. GAUGUIER, M. YERLE, F. APIOU, P. PINTON *et al.*, 1998 Analysis of distribution in the human, pig, and rat genomes points toward a general subtelomeric origin of minisatellite structures. Genomics **52:** 62-71.

ANDERSON, T. J. C., X.-Z. SU, A. RODDAM and K. P. DAY, 2000 Complex mutations in a high proportion of microsatellite loci from the protozoan parasite *Plasmodium falciparum*. Mol Ecol **9:** 1599-1608.

ANMARKRUD, J. A., O. KLEVEN, L. BACHMANN and J. T. LIFJELD, 2008 Microsatellite evolution: Mutations, sequence variation, and homoplasy in the hypervariable avian microsatellite locus HrU10. BMC Evol Biol **8:** 138.

ARCOT, S. S., Z. WANG, J. L. WEBER, P. L. DEININGER and M. A. BATZER, 1995 Alu repeats: a source for the genesis of primate microsatellites. Genomics **29:** 136-144.

BACHTROG, D., M. AGIS, M. IMHOF and C. SCHLÖTTERER, 2000 Microsatellite variability differs between dinucleotide repeat motifs-evidence from *Drosophila melanogaster*. Mol Biol Evol **17:** 1277-1285.

BACOLLA, A., J. E. LARSON, J. R. COLLINS, J. LI, A. MILOSAVLJEVIC *et al.*, 2008 Abundance and length of simple repeats in vertebrate genomes are determined by their structural properties. Genome Res **18:** 1545-1553.

BAKKER, S. C., 2005 Differences in stutter intensities between microsatellites are related to length and sequence of the repeat, pp. 81-94 in *Unravelling the genetics of schizophrenia and ADHD*, edited by S. C. BAKKER. University Medical Centre Utrecht, Utrecht.

BALLOUX, F., and N. LUGON-MOULIN, 2002 The estimation of population differentiation with microsatellite markers. Mol Ecol **11:** 155-165.

BARKER, G. C., 2002 Microsatellite DNA: a tool for population genetic analysis. Trans R Soc Trop Med Hyg **96 Suppl 1:** S21-24.

BATZER, M. A., C. M. RUBIN, U. HELLMANN-BLUMBERG, M. ALEGRIA-HARTMAN, E. P. LEEFLANG *et al.*, 1995 Dispersion and insertion polymorphism in two small subfamilies of recently amplified human *Alu* repeats. J Mol Biol **247:** 418-427.

BAYLISS, C. D., K. M. DIXON and E. R. MOXON, 2004 Simple sequence repeats (microsatellites): mutational mechanisms and contributions to bacterial pathogenesis. A meeting review. FEMS Immunol Med Microbiol **40:** 11-19.

BECK, N. R., M. C. DOUBLE and A. COCKBURN, 2003 Microsatellite evolution at two hypervariable loci revealed by extensive avian pedigrees. Molecular Biology and Evolution **20:** 54-61.

BOYER, J. C., N. A. YAMADA, C. N. ROQUES, S. B. HATCH, K. RIESS *et al.*, 2002 Sequence dependent instability of mononucleotide microsatellites in cultured mismatch repair proficient and deficient mammalian cells. Hum Mol Genet **11:** 707-713.

BROHEDE, J., C. R. PRIMMER, A. MOLLER and H. ELLEGREN, 2002 Heterogeneity in the rate and pattern of germline mutation at individual microsatellite loci. Nucleic Acids Res **30:** 1997-2003.

BROUWER, J. R., R. WILLEMSEN and B. A. OOSTRA, 2009 Microsatellite repeat instability and neurological disease. Bioessays **31:** 71-83.

BULL, L. N., C. R. PABON-PENA and N. B. FREIMER, 1999 Compound microsatellite repeats: practical and theoretical features. Genome Res **9:** 830-838.

BULUT, Z., C. R. MCCORMICK, D. GOPURENKO, R. N. WILLIAMS, D. H. BOS *et al.*, 2008 Microsatellite mutation rates in the eastern tiger salamander (*Ambystoma tigrinum tigrinum*) differ 10-fold across loci. Genetica **Online publication first**.

CASACUBERTA, E., P. PUIGDOMENECH and A. MONFORT, 2000 Distribution of microsatellites in relation to coding sequences within the *Arabidopsis thaliana* genome. Plant Sci **157:** 97-104.

CHANG, D. K., D. METZGAR, C. WILLS and C. R. BOLAND, 2001 Microsatellites in the eukaryotic DNA mismatch repair genes as modulators of evolutionary mutation rate. Genome Res **11:** 1145-1146.

COX, R., and S. M. MIRKIN, 1997 Characteristic enrichment of DNA repeats in different genomes. Proc Natl Acad Sci U S A **94:** 5237-5242.

CUMMINGS, C. J., and H. Y. ZOGHBI, 2000 Fourteen and counting: unraveling trinucleotide repeat diseases. Hum Mol Genet **9:** 909-916.

DEKA, R., M. D. SHRIVER, L. M. YU, R. E. FERRELL and R. CHAKRABORTY, 1995 Intra- and inter-population diversity at short tandem repeat loci in diverse populations of the world. Electrophoresis **16:** 1659-1664.

DETTMAN, J. R., and J. W. TAYLOR, 2004 Mutation and evolution of microsatellite Loci in neurospora. Genetics **168:** 1231-1248.

DI RIENZO, A., A. C. PETERSON, J. C. GARZA, A. M. VALDES, M. SLATKIN *et al.*, 1994 Mutational processes of simple-sequence repeat loci in human populations. Proc Natl Acad Sci U S A **91:** 3166-3170.

DIERINGER, D., and C. SCHLÖTTERER, 2003 Two distinct modes of microsatellite mutation processes: evidence from the complete genomic sequences of nine species. Genome Res **13:** 2242-2251.

EISEN, J., 1999 Mechanistic basis for microsatellite instability, pp. 34-48 in *Microsatellites: Evolution and Applications*, edited by D. B. GOLDSTEIN and C. SCHLOTTERER. Oxford University Press.

ELLEGREN, H., C. R. PRIMMER and B. C. SHELDON, 1995 Microsatellite 'evolution': directionality or bias? Nat Genet **11:** 360-362.

EVERETT, C. M., and N. W. WOOD, 2004 Trinucleotide repeats and neurodegenerative disease. Brain **127:** 2385-2405.

FALUSH, D., and Y. IWASA, 1999 Size-dependent mutability and microsatellite constraints. Molecular Biology and Evolution **16:** 960-966.

FIELD, D., and C. WILLS, 1998 Abundant microsatellite polymorphism in *Saccharomyces cerevisiae*, and the different distributions of microsatellites in eight prokaryotes and *S. cerevisiae*, result from strong mutation pressures and a variety of selective forces. Proc Natl Acad Sci U S A **95:** 1647-1652.

FONDON, J. W., 3RD, and H. R. GARNER, 2004 Molecular origins of rapid and continuous morphological evolution. Proc Natl Acad Sci U S A **101:** 18058-18063.

FRESCO, J. R., and B. M. ALBERTS, 1960 The Accommodation of Noncomplementary Bases in Helical Polyribonucleotides and Deoxyribonucleic Acids. Proc Natl Acad Sci U S A **46:** 311-321.

GALINDO, C. L., L. J. MCIVER, J. F. MCCORMICK, M. A. SKINNER, Y. XIE *et al.*, 2009 Global microsatellite content distinguishes humans, primates, animals, and plants. Mol Biol Evol.

GARGANO, G., D. CALCARA, S. CORSALE, V. AGNESE, C. INTRIVICI *et al.*, 2007 Aberrant methylation within RUNX3 CpG island associated with the nuclear and mitochondrial microsatellite instability in sporadic gastric cancers. Results of a GOIM (Gruppo Oncologico dell'Italia Meridionale) prospective study. Ann Oncol **18 Suppl 6:** vi103-109.

GREEN, H., and N. WANG, 1994 Codon reiteration and the evolution of proteins. Proc Natl Acad Sci U S A **91:** 4298-4302.

GUO, W., C. CAI, C. WANG, Z. HAN, X. SONG *et al.*, 2007 A microsatellite-based, gene-rich linkage map reveals genome structure, function, and evolution in *Gossypium*. Genetics **176:** 527-541.

GUR-ARIE, R., C. J. COHEN, Y. EITAN, L. SHELEF, E. M. HALLERMAN *et al.*, 2000 Simple sequence repeats in *Escherichia coli*: abundance, distribution, composition, and polymorphism. Genome Res **10:** 62-71.

GUYOMARD, R., S. MAUGER, K. TABET-CANALE, S. MARTINEAU, C. GENET *et al.*, 2006 A type I and type II microsatellite linkage map of rainbow trout (*Oncorhynchus mykiss*) with presumptive coverage of all chromosome arms. BMC Genomics **7:** 302.

HAMADA, H., M. G. PETRINO and T. KAKUNAGA, 1982 A novel repeated element with Z-DNA-forming potential is widely found in evolutionarily diverse eukaryotic genomes. Proc Natl Acad Sci U S A **79:** 6465-6469.

HAMADA, H., M. G. PETRINO, T. KAKUNAGA, M. SEIDMAN and B. D. STOLLAR, 1984 Characterization of genomic poly(dT-dG).poly(dC-dA) sequences: structure, organization, and conformation. Mol Cell Biol **4:** 2610-2621.

HAMMOCK, E. A., and L. J. YOUNG, 2005 Microsatellite instability generates diversity in brain and sociobehavioral traits. Science **308:** 1630-1634.

HARR, B., J. TODOROVA and C. SCHLÖTTERER, 2002 Mismatch repair-driven mutational bias in *D. melanogaster*. Mol Cell **10:** 199-205.

HARVEY, S. C., 1997 Slipped structures in DNA triplet repeat sequences: entropic contributions to genetic instabilities. Biochemistry **36:** 3047-3049.

HAVRYLIUK, T., P. ORJUELA-SANCHEZ and M. U. FERREIRA, 2008 Plasmodium vivax: microsatellite analysis of multiple-clone infections. Exp Parasitol **120:** 330-336.

HILE, S. E., and K. A. ECKERT, 2004 Positive correlation between DNA polymerase alpha-primase pausing and mutagenesis within polypyrimidine/polypurine microsatellite sequences. J Mol Biol **335:** 745-759.

HOFF-OLSEN, P., S. JACOBSEN, B. MEVAG and B. OLAISEN, 2001 Microsatellite stability in human post-mortem tissues. Forensic Sci Int **119:** 273-278.

HUANG, C. H., Y. S. LIN, Y. L. YANG, S. W. HUANG and C. W. CHEN, 1998 The telomeres of Streptomyces chromosomes contain conserved palindromic sequences with potential to form complex secondary structures. Mol Microbiol **28:** 905-916.

HUANG, Q. Y., F. H. XU, H. SHEN, H. Y. DENG, Y. J. LIU *et al.*, 2002 Mutation patterns at dinucleotide microsatellite loci in humans. Am J Hum Genet **70:** 625-634.

HURST, L. D., and H. ELLEGREN, 1998 Sex biases in the mutation rate. Trends Genet **14:** 446-452.

IRION, D. N., A. L. SCHAFFER, T. R. FAMULA, M. L. EGGLESTON and E. AL, 2003 Analysis of genetic variation in 28 dog breed populations with 100 microsatellite markers. The Journal of Heredity **94:** 81.

JAKUPCIAK, J. P., and R. D. WELLS, 1999 Genetic instabilities in (CTG.CAG) repeats occur by recombination. J Biol Chem **274:** 23468-23479.

JAKUPCIAK, J. P., and R. D. WELLS, 2000a Gene conversion (recombination) mediates expansions of CTG-CAG repeats. J Biol Chem **275:** 40003-40013.

JAKUPCIAK, J. P., and R. D. WELLS, 2000b Genetic instabilities of triplet repeat sequences by recombination. IUBMB Life **50:** 355-359.

JARNE, P., and P. J. L. LAGODA, 1996 Microsatellites, from molecules to populations and back. Trends in Ecology & Evolution **11:** 424-429.

JEFFREYS, A. J., N. J. ROYLE, V. WILSON and Z. WONG, 1988 Spontaneous mutation rates to new length alleles at tandem-repetitive hypervariable loci in human DNA. Nature **332:** 278-281.

JEFFREYS, A. J., K. TAMAKI, A. MACLEOD, D. G. MONCKTON, D. L. NEIL *et al.*, 1994 Complex gene conversion events in germline mutation at human minisatellites. Nat Genet **6:** 136-145.

JENSEN, L. E., P. A. JAUERT and D. T. KIRKPATRICK, 2005 The large loop repair and mismatch repair pathways of *Saccharomyces cerevisiae* act on distinct substrates during meiosis. Genetics **170:** 1033-1043.

JOUQUAND, S., C. PRIAT, C. HITTE, P. LACHAUME, C. ANDRE *et al.*, 2000 Identification and characterization of a set of 100 tri- and dinucleotide microsatellites in the canine genome. Animal Genetics **31:** 266-272.

JURKA, J., V. V. KAPITONOV, O. KOHANY and M. V. JURKA, 2007 Repetitive sequences in complex genomes: structure and evolution. Annu Rev Genomics Hum Genet **8:** 241-259.

KAPITONOV, V. V., A. PAVLICEK and J. JURKA, 2004 Anthology of Human Repetitive DNA in *Encyclopedia of Molecular Cell Biology and Molecular Medicine*, edited by R. A. MEYERS. Wiley-VHC, Weinheim, Germany.

KARLIN, S., L. BROCCHIERI, J. TRENT, B. E. BLAISDELL and J. MRAZEK, 2002 Heterogeneity of genome and proteome content in bacteria, archaea, and eukaryotes. Theor Popul Biol **61:** 367-390.

KATTI, M. V., P. K. RANJEKAR and V. S. GUPTA, 2001 Differential distribution of simple sequence repeats in eukaryotic genome sequences. Mol Biol Evol **18:** 1161-1167.

KATTI, M. V., R. SAMI-SUBBU, P. K. RANJEKAR and V. S. GUPTA, 2000 Amino acid repeat patterns in protein sequences: their diversity and structural-functional implications. Protein **9:** 1203-1209.

KAYSER, M., L. ROEWER, M. HEDMAN, L. HENKE, J. HENKE *et al.*, 2000 Characteristics and frequency of germline mutations at microsatellite loci from the human Y chromosome, as revealed by direct observation in father/son pairs. Am J Hum Genet **66:** 1580-1588.

KELKAR, Y. D., S. TYEKUCHEVA, F. CHIAROMONTE and K. D. MAKOVA, 2008 The genome-wide determinants of human and chimpanzee microsatellite evolution. Genome Res **18:** 30-38.

KIM, H. C., J. C. KIM, S. A. ROH, C. S. YU, J. H. YOOK *et al.*, 2005 Aberrant CpG island methylation in early-onset sporadic gastric carcinoma. J Cancer Res Clin Oncol **131:** 733-740.

KRUGLYAK, S., R. T. DURRETT, M. D. SCHUG and C. F. AQUADRO, 1998 Equilibrium distributions of microsatellite repeat length resulting from a balance between slippage events and point mutations. Proceedings of the National Academy of Sciences of the United States of America **95:** 10774-10778.

LAI, Y., and F. SUN, 2003 The relationship between microsatellite slippage mutation rate and the number of repeat units. Mol Biol Evol **20:** 2123-2131.

LEVINSON, G., and G. A. GUTMAN, 1987a High frequencies of short frameshifts in poly-CA/TG tandem repeats borne by bacteriophage M13 in *Escherichia coli* K-12. Nucleic Acids Res **15:** 5323-5338.

LEVINSON, G., and G. A. GUTMAN, 1987b Slipped-strand mispairing: a major mechanism for DNA sequence evolution. Mol Biol Evol **4:** 203-221.

LI, X. L., Y. F. GONG, J. W. ZHANG, Z. Z. LIU and A. VALENTINI, 2004 Study on polymorphisms of microsatellites DNA of six Chinese indigenous sheep breeds. Yi Chuan Xue Bao **31:** 1203-1210.

LIM, S., L. NOTLEY-MCROBB, M. LIM and D. A. CARTER, 2004 A comparison of the nature and abundance of microsatellites in 14 fungal genomes. Fungal Genet Biol **41:** 1025-1036.

LIN, X., and T. ASHIZAWA, 2005 Recent progress in spinocerebellar ataxia type-10 (SCA10). Cerebellum **4:** 37-42.

LITT, M., and J. A. LUTY, 1989 A hypervariable microsatellite revealed by in vitro amplification of a dinucleotide repeat within the cardiac muscle actin gene. Am J Hum Genet **44:** 397-401.

MAKOVA, K. D., and W. H. LI, 2002 Strong male-driven evolution of DNA sequences in humans and apes. Nature **416:** 624-626.

MALPERTUY, A., B. DUJON and G. F. RICHARD, 2003 Analysis of microsatellites in 13 hemiascomycetous yeast species: mechanisms involved in genome dynamics. J Mol Evol **56:** 730-741.

MAR ALBÀ, M., M. F. SANTIBÁÑEZ-KOREF and J. M. HANCOCK, 1999 Amino acid reiterations in yeast are overrepresented in particular classes of proteins and show evidence of a slippage-like mutational process. J Mol Evol **49:** 789-797.

MCLEAN, J. E., T. R. SEAMONS, M. B. DAUER, P. BENTZEN and T. P. QUINN, 2008 Variation in reproductive success and effective number of breeders in a hatchery population of

steelhead trout (*Oncorhynchus mykiss*): examination by microsatellite-based parentage analysis Conservation Genetics **9:** 295-304.

METZGAR, D., J. BYTOF and C. WILLS, 2000 Selection against frameshift mutations limits microsatellite expansion in coding DNA. Genome Res **10:** 72-80.

MLAMBO, G., D. SULLIVAN, S. L. MUTAMBU, W. SOKO, J. MBEDZI *et al.*, 2007 Analysis of genetic polymorphism in select vaccine candidate antigens and microsatellite loci in *Plasmodium falciparum* from endemic areas at varying altitudes. Acta Trop **102:** 201-205.

MONTOYA, L., M. GALLEGO, B. GAVIGNET, R. PIARROUX, J. A. RIOUX *et al.*, 2007 Application of microsatellite genotyping to the study of a restricted *Leishmania infantum* focus: different genotype compositions in isolates from dogs and sand flies. Am J Trop Med Hyg **76:** 888-895.

MORGANTE, M., M. HANAFEY and W. POWELL, 2002 Microsatellites are preferentially associated with nonrepetitive DNA in plant genomes. Nat Genet **30:** 194-200.

MOXON, E. R., P. B. RAINEY, M. A. NOWAK and R. E. LENSKI, 1994 Adaptive evolution of highly mutable loci in pathogenic bacteria. Curr Biol **4:** 24-33.

MOXON, R., C. BAYLISS and D. HOOD, 2006 Bacterial contingency loci: the role of simple sequence DNA repeats in bacterial adaptation. Annu Rev Genet **40:** 307-333.

NADIR, E., H. MARGALIT, T. GALLILY and S. A. BEN-SASSON, 1996 Microsatellite spreading in the human genome: evolutionary mechanisms and structural implications. Proc Natl Acad Sci U S A **93:** 6470-6475.

NEFF, B. D., P. FU and M. R. GROSS, 1999 Microsatellite evolution in sunfish (Centrarchidae). Canadian Journal of Fisheries and Aquatic Sciences **56:** 1198-1205.

NEFF, B. D., and M. R. GROSS, 2001 Microsatellite evolution in vertebrates: Inference from AC dinucleotide repeats. Evolution **55:** 1717-1733.

OKADA, A., and H. B. TAMATE, 2000 Pedigree Analysis of the Sika Deer (Cervus nippon) using Microsatellite Markers. Zoolog Sci **17:** 335-340.

ORR, H. T., and H. Y. ZOGHBI, 2007 Trinucleotide repeat disorders. Annu Rev Neurosci **30:** 575-621.

RHODES, M., R. STRAW, S. FERNANDO, A. EVANS, T. LACEY *et al.*, 1998 A high-resolution microsatellite map of the mouse genome. Genome Res **8:** 531-542.

RICHARD, G.-F., A. KERREST and B. DUJON, 2008 Comparative Genomics and Molecular Dynamics of DNA Repeats in Eukaryotes. Microbiol. Mol. Biol. Rev. **72:** 686-727.

RICHARD, G. F., G. M. GOELLNER, C. T. MCMURRAY and J. E. HABER, 2000 Recombination-induced CAG trinucleotide repeat expansions in yeast involve the MRE11-RAD50-XRS2 complex. Embo J **19:** 2381-2390.

RICHARD, G. F., and F. PAQUES, 2000 Mini- and microsatellite expansions: the recombination connection. EMBO Rep **1:** 122-126.

ROCKMAN, M. V., and G. A. WRAY, 2002 Abundant raw material for cis-regulatory evolution in humans. Mol Biol Evol **19:** 1991-2004.

RÖDER, M. S., V. KORZUN, K. WENDEHAKE, J. PLASCHKE, M.-H. TIXIER *et al.*, 1998 A microsatellite map of wheat. Genetics **149:** 2007-2023.

ROSE, O., and D. FALUSH, 1998 A threshold size for microsatellite expansion. Mol Biol Evol **15:** 613-615.

SAGHER, D., A. HSU and B. STRAUSS, 1999 Stabilization of the intermediate in frameshift mutation. Mutat Res **423:** 73-77.

SAINUDIIN, R., R. T. DURRETT, C. F. AQUADRO and R. NIELSEN, 2004 Microsatellite mutation models: Insights from a comparison of humans and chimpanzees. Genetics **168:** 383-395.

SAKAMOTO, N., J. E. LARSON, R. R. IYER, L. MONTERMINI, M. PANDOLFO *et al.*, 2001 GGA*TCC-interrupted triplets in long GAA*TTC repeats inhibit the formation of triplex and sticky DNA structures, alleviate transcription inhibition, and reduce genetic instabilities. J Biol Chem **276:** 27178-27187.

SCHLÖTTERER, C., 2000 Evolutionary dynamics of microsatellite DNA. Chromosoma **109:** 365-371.

SCHLÖTTERER, C., and D. TAUTZ, 1992 Slippage synthesis of simple sequence DNA. Nucleic Acids Res **20:** 211-215.

SEYFERT, A. L., M. E. CRISTESCU, L. FRISSE, S. SCHAACK, W. K. THOMAS *et al.*, 2008 The rate and spectrum of microsatellite mutation in *Caenorhabditis elegans* and *Daphnia pulex*. Genetics **178:** 2113-2121.

SHAH, S. N., S. E. HILE and K. A. ECKERT, 2010 Defective Mismatch Repair, Microsatellite Mutation Bias, and Variability in Clinical Cancer Phenotypes. Cancer Res.

SHINDE, D., Y. LAI, F. SUN and N. ARNHEIM, 2003 Taq DNA polymerase slippage mutation rates measured by PCR and quasi-likelihood analysis: (CA/GT)n and (A/T)n microsatellites. Nucleic Acids Res **31:** 974-980.

SIA, E. A., R. J. KOKOSKA, M. DOMINSKA, P. GREENWELL and T. D. PETES, 1997 Microsatellite instability in yeast: dependence on repeat unit size and DNA mismatch repair genes. Mol Cell Biol **17:** 2851-2858.

SMITH, G. P., 1976 Evolution of repeated DNA sequences by unequal crossover. Science **191:** 528-535.

SREENU, V. B., P. KUMAR, J. NAGARAJU and H. A. NAGARAJARAM, 2006 Microsatellite polymorphism across the *M. tuberculosis* and *M. bovis* genomes: implications on genome evolution and plasticity. BMC Genomics **7:** 78.

STRAND, M., T. A. PROLLA, R. M. LISKAY and T. D. PETES, 1993 Destabilization of tracts of simple repetitive DNA in yeast by mutations affecting DNA mismatch repair. Nature **365:** 274-276.

STRAUSS, B. S., 1999 Frameshift mutation, microsatellites and mismatch repair. Mutat Res **437:** 195-203.

STRAUSS, B. S., D. SAGHER and S. ACHARYA, 1997 Role of proofreading and mismatch repair in maintaining the stability of nucleotide repeats in DNA. Nucleic Acids Res **25:** 806-813.

STREISINGER, G., and J. OWEN, 1985 Mechanisms of spontaneous and induced frameshift mutation in bacteriophage T4. Genetics **109:** 633-659.

SUBRAMANIAN, S., R. K. MISHRA and L. SINGH, 2003 Genome-wide analysis of microsatellite repeats in humans: their abundance and density in specific genomic regions. Genome Biol **4:** R13.

SUN, W., H. CHEN, C. LEI, X. LEI and Y. ZHANG, 2008 Genetic variation in eight Chinese cattle breeds based on the analysis of microsatellite markers. Genet Sel Evol **40:** 681-692.

SUNDSTROM, H., M. T. WEBSTER and H. ELLEGREN, 2003 Is the rate of insertion and deletion mutation male biased?: Molecular evolutionary analysis of avian and primate sex chromosome sequences. Genetics **164:** 259-268.

SYMONDS, V. V., and A. M. LLOYD, 2003 An analysis of microsatellite loci in *Arabidopsis thaliana*: Mutational dynamics and application. Genetics **165:** 1475-1488.

TAFT, R. J., M. PHEASANT and J. S. MATTICK, 2007 The relationship between non-protein-coding DNA and eukaryotic complexity. Bioessays **29:** 288-299.

TAKEZAKI, N., and M. NEI, 2009 Genomic drift and evolution of microsatellite DNAs in human populations. Mol Biol Evol.

TAUTZ, D., M. TRICK and G. A. DOVER, 1986 Cryptic simplicity in DNA is a major source of genetic variation. Nature **322:** 652-656.

TEMNYKH, S., G. DECLERCK, A. LUKASHOVA, L. LIPOVICH, S. CARTINHOUR *et al.*, 2001 Computational and experimental analysis of microsatellites in rice (*Oryza sativa* L.): frequency, length variation, transposon associations, and genetic marker potential. Genome Res **11:** 1441-1452.

THUILLET, A. C., D. BRU, J. DAVID, P. ROUMET, S. SANTONI *et al.*, 2002 Direct estimation of mutation rate for 10 microsatellite loci in durum wheat, *Triticum turgidum* (L.) Thell. ssp durum desf. Mol Biol Evol **19:** 122-125.

TOTH, G., Z. GASPARI and J. JURKA, 2000 Microsatellites in different eukaryotic genomes: survey and analysis. Genome Res. **10:** 967-981.

TSUTSUI, N. D., A. V. SUAREZ, D. A. HOLWAY and T. J. CASE, 2000 Reduced genetic variation and the success of an invasive species. Proc Natl Acad Sci U S A **97:** 5948-5953.

TYSON, J., and J. C. MATHERS, 2007 Dietary and genetic modulation of DNA repair in healthy human adults. Proc Nutr Soc **66:** 42-51.

UMAR, A., J. C. BOYER, D. C. THOMAS, D. C. NGUYEN, J. I. RISINGER *et al.*, 1994 Defective mismatch repair in extracts of colorectal and endometrial cancer cell lines exhibiting microsatellite instability. J Biol Chem **269:** 14367-14370.

VAN BELKUM, A., S. SCHERER, L. VAN ALPHEN and H. VERBRUGH, 1998 Short-sequence DNA repeats in prokaryotic genomes. Microbiol Mol Biol Rev **62:** 275-293.

VAN TREUREN, R., H. KUITTINEN, K. KARKKAINEN, E. BAENA-GONZALEZ and O. SAVOLAINEN, 1997 Evolution of microsatellites in *Arabis petraea* and *Arabis lyrata*, outcrossing relatives of *Arabidopsis thaliana*. Mol Biol Evol **14:** 220-229.

VERSTREPEN, K. J., A. JANSEN, F. LEWITTER and G. R. FINK, 2005 Intragenic tandem repeats generate functional variability. Nat Genet **37:** 986-990.

VIARD, F., P. BREMOND, R. LABBO, F. JUSTY, B. DELAY *et al.*, 1996 Microsatellites and the genetics of highly selfing populations in the freshwater snail *Bulinus truncatus*. Genetics **142:** 1237-1247.

WALTER, R., and B. K. EPPERSON, 2004 Microsatellite analysis of spatial structure among seedlings in populations of *Pinus strobus* (Pinaceae). Am. J. Bot. **91:** 549-557.

WEBER, J. L., and C. WONG, 1993 Mutation of human short tandem repeats. Hum Mol Genet **2:** 1123-1128.

WEBSTER, M. T., N. G. C. SMITH and H. ELLEGREN, 2002 Microsatellite evolution inferred from human-chimpanzee genomic sequence alignments. Proceedings of the National Academy of Sciences of the United States of America **99:** 8748-8753.

WELCH, J. W., D. H. MALONEY and S. FOGEL, 1990 Unequal crossing-over and gene conversion at the amplified CUP1 locus of yeast. Mol Gen Genet **222:** 304-310.

WERNER, P., C. S. MELLERSH, M. G. RADUCHA, S. DEROSE, G. M. ACLAND *et al.*, 1999a Anchoring of canine linkage groups with chromosome-specific markers. Mamm Genome **10:** 814-823.

WERNER, P., M. G. RADUCHA, U. PROCIUK, P. S. HENTHORN and D. F. PATTERSON, 1999b A comparative approach to physical and linkage mapping of genes on canine chromosomes using gene-associated simple sequence repeat polymorphisms illustrated by studies of dog chromosome 9. J Hered **90:** 39-42.

WIERDL, M., M. DOMINSKA and T. D. PETES, 1997 Microsatellite instability in yeast: dependence on the length of the microsatellite. Genetics **146:** 769-779.

WONG, G. K., D. A. PASSEY and J. YU, 2001 Most of the human genome is transcribed. Genome Res **11:** 1975-1977.

WRIGHT, J., 1994 Mutation at VNTRs: Are minisatellites the evolutionary progeny of microsatellites? Genome **37:** 345-347.

XU, X., M. PENG and Z. FANG, 2000 The direction of microsatellite mutations is dependent upon allele length. Nat Genet **24:** 396-399.

ZAJC, I., and J. SAMPSON, 1996 DNA microsatellites in domesticated dogs: application in paternity disputes. Pflugers Arch **431:** R201-202.

ZHANG, D., B. BOCCARA, L. MOTILAL, D. R. BUTLER, P. UMAHARAN *et al.*, 2008 Microsatellite variation and population structure in the "Refractario" cacao of Ecuador Conservation Genetics **9:** 327-337.

ZHU, Y., D. C. QUELLER and J. E. STRASSMANN, 2000a A phylogenetic perspective on sequence evolution in microsatellite loci. Journal of Molecular Evolution **50:** 324-338.

ZHU, Y., J. E. STRASSMANN and D. C. QUELLER, 2000b Insertions, substitutions, and the origin of microsatellites. Genetical Research **76:** 227-236.

CHAPTER II: Finding Microsatellites within Genomes: Algorithmic Biases and Conflicting Definitions

Abstract

In this chapter I deal with a fundamental problem for bioinformatic research on microsatellites and other tandem repeats. I performed a comprehensive and critical review of algorithms and programs designed to find tandem repeats within DNA sequences, focusing on their suitability for *de novo* identification of microsatellites within whole genomic sequences. I also researched the factors which influence the comparability of results from different programs and, therefore, across different studies. For these purposes, I carried out comparisons among 19 microsatellite search programs featuring suitable characteristics for genomic microsatellite searches I conclude that the best, although not ideal, search results can be obtained with the programs SciRoKo and Tandem Repeat Finder (TRF). In addition, I present some criticism on the measures used to express microsatellite results, both in informatics and comparative genomics studies, which usually flaw present and future comparisons of the data. Based on my observations and analysis I suggest a set of basic informative criterions to be included in publications involving microsatellite searches, in order to significantly improve the usefulness of published results in future research.

2.1 Introduction

Genome sequences, especially in eukaryotes, attain very large sizes in the order of billions of characters, and are very rich in repetitive regions. Microsatellites are the repetitive sequences with the shortest repeat units in DNA, defined by convention to be one to six nucleotides in length. Despite this relatively short size, microsatellites can have a great influence within genomes because they are highly abundant and have the highest mutation rate among DNA components (ELLEGREN 2004). The large size of genomic sequences (e.g. ~50 to 700 millions of nucleotides for animal chromosomes) coupled with the extremely high occurrence of microsatellites within these, renders the detection of microsatellites within genomic sequences as a particularly demanding task requiring highly efficient computer programs.

From an informatics point of view, microsatellites and other repetitive segments within DNA sequences are nuisances, because comparisons of DNA sequences and searches within DNA databases can be flawed or seriously complicated by the presence of repeats. Therefore, the first efforts to identify these sequences and to register their positions were done with the objective to exclude them from further analyses (see the RepeatMasker webpage: http://www.repeatmasker.org/). The interest in finding tandem repeats, especially microsatellites, for their own sake, grew when large high quality sequences corresponding to genome assemblies started to become available.

Microsatellites are very useful as molecular markers, but their identification by traditional wet lab procedures is complicated and time-consuming (reviewed by ZANE *et al.* 2002). Therefore, the possibility of retrieving the positions and sequence information for microsatellites directly from DNA databases offered a great advantage. Furthermore, the generation of repeats, both tandem and dispersed, is an important evolutionary process within genomes (HAUBOLD and WIEHE 2006), and it is becoming widely accepted that these repeats can have a potpourri of functions and effects (reviewed by RICHARD *et al.* 2008). Therefore, the challenge of detecting repeats efficiently in large DNA sequences became on vogue among computer scientists during the last ten years.

An important issue when analyzing DNA repeats is that these are biological features subject to dynamic changes throughout time. Therefore, individual repeat units may contain imperfections or interruptions (nucleotides which do not match the pattern) degrading the similarity among units. Tandem repeats containing imperfections have been referred to in

bioinformatics' lingo as "approximate tandem repeats" ATRs (see DE RIDDER *et al.* 2006; WEXLER *et al.* 2005) and, in some cases, also classified into different categories of approximation (HAUTH and JOSEPH 2002). The efficient detection of tandem repeats in large DNA sequences, allowing for a certain degree of imperfection within repeat units, as long as these do not disturb the overall periodicity of the repeat, soon became part of the requirements for general purpose repeat finders.

At present, there is an extensive choice of programs to search for tandem repeats, both with perfect and approximate repeat units, which should cover the current demand. A recent review by Sharma and collaborators (2007) presents a representative summary of repeat finding software, most of it specific for microsatellites. This large quantity of programs (20+ programs as of 2008), together with the constantly increasing processing power of computers, should provide an excellent base to carry out microsatellite searches within genomes. However, during my survey of these algorithms I noticed several critical flaws which may in great part be attributed to a lack of communication between computer scientists and biologists. Many programs have not been published in scientific publications and/or lack informative descriptions, and most of them are missing adequate benchmarking. Moreover, the advantages of using the programs are pompously advertised in web pages or publications, but the possible shortcomings of these tools are left for the user to find out.

A recent publication by Leclercq *et al.* (2007) presents a comparison of search results among five repeat finding programs (TRF, Sputnik, mreps, STAR, and RepeatMasker). The authors show that results produced by these programs with several sets of parameter combinations can vary to a significant extent. However, these comparisons are blemished because the authors missed the important point, due to the marked differences among these five algorithms, the input parameters should be fine-tuned individually for each program so that the searches become equivalent. No significant comparisons can be carried out if the programs are not looking for the same characteristics. Some publications presenting new repeat finding programs do present output comparisons with other programs, for example see the publications for TROLL (CASTELO *et al.* 2002), STAR (DELGRANGE and RIVALS 2004), ATRHunter (WEXLER *et al.* 2005), and tandem (DOMANIC and PREPARATA 2007). However, the same criticism as above applies to these studies, and it is striking that the main criteria for the comparison of programs are overall execution time and the total number of hits obtained; the less execution time and the more hits, the better the repeat finding program.

As will be shown throughout my thesis, the definition of microsatellites is rather complex, and the interpretation of search results for these sequences needs to be adjusted accordingly. In the present Chapter I start by carrying out a critical analysis and comparison of tandem repeat finding programs suitable for microsatellite detection, with emphasis on programs with the capacity to analyze large sequences in the order of millions of nucleotides. I further define some guidelines for the analysis and comparison of tandem repeat finder results. Finally I select programs with the best features for the generation of unbiased microsatellite databases for a comparative analysis of microsatellite abundance and distribution among large genomes.

2.1.1 Pattern discovery: Detecting repeats in DNA sequences

The search of repeated patterns within DNA sequences is among the most important and basic pattern discovery techniques in computational biology. A DNA repeat is a substring of nucleotides that occurs two or more times within a DNA string. Corresponding repeat units can be close to each other or at different locations in a sequence. When two or more repeats occur in a head to tail fashion they constitute a tandem repeat, and if the repeat units are one to six nucleotides long, the tandem repeat is classified as a microsatellite (for more details about DNA repeats see section 1.3). In the case of DNA, the characters in the string can be **A**, **G**, **C**, **T,** or **a, g, c, t,** representing known nucleotides, and **N** or **n** representing unknown nucleotides. The specific nucleotide sequence of a repetitive unit is called a **motif**.

DNA repeats are usually not perfect repetitions of a motif, but suffer modifications in their sequence during biological processes. These can basically involve the substitution of one nucleotide for another (**point mutations**), or insertions and deletions (**indels**) of one or more nucleotides. Therefore, repeat units are not always identical and can show various degrees of dissimilarity (degradation). In the case of microsatellites, a combination of these modifications with the typical expansions and contractions of microsatellites, can lead to complex patterns.

To program a repeat finder, the repeat or repeats to look for have to be defined to the program. Based on this definition, a pattern matching algorithm can be used to index all occurrences that match the definition. If the nucleotide composition of the repeat is known, it is "literally" defined by giving the possible repeat motifs as input in a list (i.e. AT, AC, AG, CG, CTG, etc.). This is the case of the programs TROLL (CASTELO *et al.* 2002) and

RepeatMasker (SMIT *et al.* 1996-2007), which use a **"dictionary approach"** to find all exact or approximate matches to the specific queries, respectively. TROLL is a perfect microsatellite finder, meaning that it only looks for perfect repetitions of motifs of length 1 to 6 nucleotides. This kind of query is relatively easy to define exactly to the program by putting together all permutations for words one to six nucleotides to be formed with the four basic DNA nucleotides ($4^1+4^2+4^3+4^4+4^5+4^6$= 5460 motif combinations). In the case of RepeatMasker, a program aimed at finding repeats with motifs of any size, this kind of definition is nearly impossible. RepeatMasker works by searching the query sequence for known and consensus repeats (including nucleotide wildcards) stored in a library (Repbase, JURKA *et al.* 1992). This library needs to be constructed beforehand and it is species-specific, because many large repeats (e.g. transposons, RNA repeats) have developed or diverged differently in different species (JURKA 1998). The repeat library of RepeatMasker is far from exhaustive and it will not find all repeats in a sequence. An exhaustive search for repeats was never the aim for the RepeatMasker program, although this is sometimes believed to be the case by biologists. Therefore, it is not considered as a "repeat finder" *per se* in computer science (KURTZ *et al.* 2000).

A "real" repeat finder should be able to detect repeats in a DNA sequence without *a priori* knowledge of what the repeat units will look like (BENSON 1999). It should also have no restrictions as to the motif, motif length, and the number of copies that can be detected. The most straight-forward and exhaustive way to do this is by aligning the query sequence to itself. However, the processing time for this kind of alignment increases in an exponential fashion with the sequence length and the amount of repeats in the sequence (PARISI *et al.* 2003). Therefore, diverse strategies have been designed for the *de novo* detection of repeats more time-efficiently, by combining data structures like suffix trees (DELCHER *et al.* 1999) with fast gapped-alignment programs like WUBLAST (GISH 1996-2007) (e.g. REPuter (KURTZ and SCHLEIERMACHER 1999), RepeatFinder (VOLFOVSKY *et al.* 2001), FORRepeats (LEFEBVRE *et al.* 2002

), Tallymer (KURTZ *et al.* 2008)). Another possibility is avoiding sequence alignments and counting oligonucleotide (word) frequencies instead. By constructing clusters of "oligonucleotide excess probability clouds" obtained from the query sequence, repeat regions can be identified (GU *et al.* 2008). Numerous other *de novo* repeat finding programs have been published almost simultaneously in recent years; a representative list of all these repeat finders is shown in **table 2.1**, but their specific description is outside the scope of this

document (for a recent review and comparison of six of these programs see SAHA *et al.* 2008).

Table 2.1: Repeat finders not specific for microsatellite searches

Year	Program	Webpage	Publication
1996	RepeatMasker	http://www.repeatmasker.org/	(SMIT *et al.* 1996-2007)
1998	TEIRESISAS	__	(RIGOUTSOS and FLORATOS 1998)
1999	Reputer	http://bibiserv.techfak.uni-bielefeld.de/reputer/	(KURTZ and SCHLEIERMACHER 1999)
2001	RepeatFinder	ftp://ftp.tigr.org/pub/software/repeatFinder/	(VOLFOVSKY *et al.* 2001)
2002	FORRepeats	http://al.jalix.org/FORRepeats/	(LEFEBVRE *et al.* 2002)
2005	Recon	http://www.genetics.wustl.edu/eddy/recon	(PRICE *et al.* 2005), (BAO and EDDY 2002)
2005	PILER	http://www.drive5.com/piler/	(EDGAR and MYERS 2005)
2005	ReAS	__	(LI *et al.* 2005)
2006	Spectral Repeat Finder (SRF)	http://www.imtech.res.in/raghava/srf/	(SHARMA *et al.* 2004)
2006	WindowMasker	ftp://ftp.ncbi.nih.gov/toolbox/ncbi_tools++/CURRENT/	(MORGULIS *et al.* 2006)
2007	SERV	http://hulsweb1.cgr.harvard.edu/SERV/	(LEGENDRE *et al.* 2007)
2008	Tallymer	http://www.zbh.uni-hamburg.de/Tallymer	(KURTZ *et al.* 2008)

The most abundant subtype of repeats within DNA are tandem repeats, where two or more copies of a repeat motif are arranged contiguously in a head to tail fashion. Tandem repeats can be grouped into microsatellites, minisatellites, and satellites, based on their repeat unit size (see Chapter I). A search for tandem repeats is easier to define than a search for interspersed repeats, because it is restricted to direct adjacent repetitions of a motif. Based on this expectation and sometimes also restricting the length of the motif, a mathematical model can be constructed to predict tandem repeats (see KANNAN and MYERS 1996; LANDAU *et al.* 1987; SAGOT and MYERS 1998; SCHMIDT 1998). Once a putative tandem repeat for a specific motif has been determined, its validity is assessed by comparing it with a consensus version of the motif or the whole repeat (called consensus because in the case of imperfect repeats more than one motif may be detected, and a consensus repeat is

determined based on the possible origin of the repeat). This similarity is usually assessed by using distance measures, the two basic ones being the Hamming distance and the edit distance.

The **Hamming distance** is defined as the number of base mismatches between two strings of equal length. It is based on a one to one comparison of corresponding nucleotides and can therefore not handle indels which would produce misalignments. The string TAGTAGTACTAG, for example, is a trinucleotide repeat with four repeats and, by comparison with its idealized perfect counterpart (consensus repeat), it has a Hamming distance of 1. The fact that the Hamming distance can not deal with indels reduces its usefulness for DNA sequence comparison. This distance is, however, easy and fast to compute and was therefore used in several repeat finding programs (KOLPAKOV and KUCHEROV 2001; LANDAU *et al.* 2001; SIM *et al.* 1999).

The **edit distance**, or Levenshtein metric, is defined as the minimum number of insertions, deletions, and substitutions necessary to transform one substring into the other (BENSON 1999; DOMANIC and PREPARATA 2007). In the case of the sequence above containing an additional indel, TAGTAGGTACTAG, the edit distance in comparison with its corresponding consensus tandem repeat is 2. A more specialized version of edit distance is the **scoring matrix**, which can be viewed as a 'weighted edit distance' (SOKOL *et al.* 2007). In this case insertions, deletions and substitutions are given distinct values. Because indels produce misalignments between the putative and model tandem repeats, local sequence alignments are necessary to compute the edit distances. The computation of local alignments and edit distance calculations is a complex and time-consuming process because the number of possible local alignments and the number of possible consensus words increase in an exponential fashion with the length of the sequence. Therefore, execution time has been and, to some extent, continues being the main issue when writing and comparing tandem repeat search algorithms (see the computational complexity comparisons in DOMANIC and PREPARATA 2007). The informatics answer to these computational complexity issues is recurring to heuristics (also called approximate methods). A heuristic algorithm implements intelligent search strategies to avoid the need for exhaustive searches, so to reduce the running time for the search, while still giving a pretty good search result.

A very common strategy for heuristic programs is to analyze the sequence divided into windows, to perform the comparison among windows instead of among individual

sequence characters (CAMPAGNA *et al.* 2005). These windows can be partially overlapping or adjacent in the sequence. Obviously, the longer the window the fewer windows will need to be analyzed. However, the analysis of longer windows also requires more time, and it reduces the sensitivity of the algorithm for detecting shorter tandem repeats. On the other hand, smaller windows can also identify a huge amount of small redundant hits in the case of very long tandem repeats, which would then require a considerable amount of filtering. Therefore, the selection of the window size is crucial for the adequate functioning of the algorithm, and it needs to be determined depending on the motif length of the tandem repeats to be detected.

The most efficient repeat finding programs are those based on heuristics like the aforementioned window-based sequence comparison. These programs are divided in two phases: screening phase and verification phase. The heuristic search method is applied during the "screening phase" to reduce the amount of candidate repeat hits to those with a high probability of being a real repeat. The selected hits are used as seeds for repeat expansion during the "verification phase". Search methods based on heuristics are usually called **approximate search methods,** irrespective of the kind of repeats they screen for, because the result constitutes the best possible approximation to the real repeat content of a genome. Some examples of programs based on heuristic approaches are TRF (BENSON 1999), STRING (PARISI *et al.* 2003), tandem (DOMANIC and PREPARATA 2007), and ATRHunter (WEXLER *et al.* 2005).

A different strategy for finding repetitive regions within a sequence was published by Milosavljevic and Jurka (1993). They presented a method to discover "significantly simple sequences" or DNA sequences that contain repeated occurrences of certain "words". A stretch of sequence composed of tandem repeats, which may also contain some imperfections or interruptions, can be coded in a space-efficient way by representing it as a set of characteristics instead of the whole raw sequence. The stored characteristics are the motif, the number of repeats, and a list of imperfections (indels, substitutions) with their respective positions. The more repeats the sequence contains, the more efficient the compression algorithm will be and therefore the significance of each detected pattern is measured by the compression rate of its sequence. This is the principle used to compress files in a computer, where redundant fragments of strings of data can be stored in less space than those which are random, and algorithms incorporating this principle are called **compression algorithms** (see also DELGRANGE *et al.* 1999; DELGRANGE and RIVALS 2004; RIVALS *et al.* 1997)

The first algorithms developed for finding tandem repeats were **exact algorithms**, which attempt to carry out exhaustive searches for the tandem repeats contained in a sequence. The application of these algorithms was limited to relatively short sequences (i.e. 100000 nt) because these methods were not very efficient (BENSON and WATERMAN 1994; SCHMIDT 1998), in part due to the low processing capacity of computer processors at the time. When larger DNA sequences corresponding to whole chromosomes became available the trend in repeat finder programming switched towards heuristics in order to maximize the use of information available. The data processing efficiency of computers is, however, constantly increasing, and new algorithms for the exhaustive search of tandem repeats in large DNA sequences have more recently become available: Perfect Tandem Repeat Finding Executable (ptrfinder COLLINS *et al.* 2003), Tandem Repeat Software (TRED SOKOL *et al.* 2007), and Phobos(MAYER 2007). A selection of common and not so common algorithmic strategies developed during the last fifteen years to tackle the complex problem of finding tandem repeats is presented below.

2.1.2 Microsatellite search programs

In this section I review the most important tandem repeat finders available in the literature, which have the potential to be useful for whole genome microsatellite searches. The descriptions of the programs are in chronological order of publication and include a summary of information required to run the program. Unless stated otherwise, the programs accept FASTA format as input sequences.

Tandyman

This is a program written in Perl to find perfect tandem DNA repeats in entire genomes. The authors are Robert W. Leach and Catherine Cleland from the Los Alamos National Laboratory. The program has been available since 1997 and can be run online at the Tandyman webpage (http://hemisphere.lanl.gov/tandyman/cgi-bin/tandyman.cgi) or downloaded and run standalone on a console with Perl 5 installed (it requires the Perl module `Getopt::Long`). There is no journal publication for this program. The program offers several input options shown in **figure 2.1**, and the output is a tab delimited file containing the repeat start and end coordinates, the motif, and the number of repetitions.

```
Usage:       tandyman.pl -i sequence_file
               REQUIRED:
                 -i <fasta sequence file>
               OPTIONAL:
                 -c <coordinates file>
                 -u <repeat unit size upper limit> (default: 1/2 sequence)
                 -l <repeat unit size lower limit> (default: 2)
                 -m <minimum number of units in a repeat>
                 -e turns off sequence error checking
                 -p <permissible characters to check sequence with
                     (default: ATCGBDHVRYKMSWN), case insensitive>
                      WARNING: Reverse complimenting will not happen if you
                               use this option
                 -r reports unit coordinates instead of repeat coordinates
                 -g no reverse complimenting within backwards coordinates
                 -s reports status of progress through standard error
                    output by current unit size for which it is searching
```

Figure 2.1: Tandyman usage

TRF (Tandem Repeat Finder)

This is a program written by Gary Benson from the Department of Biomatematical Sciences at the Mount Sinai School of Medicine. It is useful to search for perfect and interrupted tandem repeats in DNA sequences without the need to specify the motif or motif length, and it can look for repeats with motifs of up to 2000 nucleotides in length. The algorithm is divided in two main phases or components: a detection component and an analysis component. The detection component uses a heuristic approach to find candidate tandem repeats based on the detection of *k-tuple* matches in order to avoid the need for full scale alignment matrix computations, as done for sequence alignments (BENSON and WATERMAN 1994). A *k-tuple* is a run of *k* consecutive characters from the nucleotide sequence, and a *k-tuple* match is a *k-tuple* with the same sequence as the *k-tuple* it is compared to. The detection criteria are based on a stochastic model of probabilities of matches and indels. Also, it treats substitutions and indels separately, and the penalties for these can be modified by the user (BENSON 1999). Because it is based on a probabilistic model, the gap penalties are length dependent.

In the analysis component, sequence alignments using Wraparound Dynamic Programming are performed for all positive candidate tandem repeats. These are aligned to perfect versions of a candidate motif and, if at least two copies of the motif can be aligned, the repeat can be reported. The final set of hits reported will depend on the parameters specified by the user to filter out non-significant hits.

An earlier version of this algorithm required the input of the desired motif size to start a search (BENSON and WATERMAN 1994), but the current program (TRF 4.00) does not require *a*

priori knowledge of the motif size or number of copies of the repeats. Command line and graphical interface versions of TRF are available for Linux and Windows. The usage and available parameters are shown in **figure 2.2**

```
trf400.dos.exe File Match Mismatch Delta PM PI Minscore MaxPeriod [options]

Where: (all weights, penalties, and scores are positive)
  File = sequences input file
  Match  = matching weight
  Mismatch  = mismatching penalty
  Delta = indel penalty
  PM = match probability (whole number)
  PI = indel probability (whole number)
  Minscore = minimum alignment score to report
  MaxPeriod = maximum period size to report
  [options] = one or more of the following :
              -m    masked sequence file
              -f    flanking sequence
              -d    data file
              -h    suppress html output
```

Figure 2.2: TRF usage

SSR screener

This is a program written in C by C.J. Cohen (cyril@tx.technion.ac.il). No paper has been published describing the program, but it was applied in a microsatellite distribution study in *Escherichia coli* by Gur-Arie *et al.* (2000). The program seems to report only perfect microsatellites with motifs from 1 to 10 nucleotides in length. The minimum number of repeats or minimum length for the microsatellite hits can be specified by the user after invoking the program by invoking the executable file from a DOS console.

MISA (Microsatellite Identification Tool)

This is a Perl script written by Thomas Thiel from the Plant Genome Resources Center (PGRC) (unpublished). It is useful for finding microsatellites with perfect as well as interrupted (compound) tandem repeats, where the amount of interruptions is determined by a maximum inter-microsatellite gap given as input by the user. It includes supplemental Perl modules to facilitate the pre-processing of input sequences and the design of primers from the program's output using primer3 (a program developed by ROZEN and SKALETSKY 1998). MISA has been used often for the development of microsatellite markers from ESTs databases (CERESINI *et al.* 2005; KOTA *et al.* 2001; PALMIERI *et al.* 2007; THIEL *et al.* 2003), to analyze microsatellite distribution (see GROVER *et al.* 2007) and to construct microsatellite databases (AISHWARYA *et al.* 2007; AISHWARYA and SHARMA 2007)

The Perl scripts for MISA and its supplementary tools can be downloaded from the MISA web page: http://pgrc.ipk-gatersleben.de/misa/ (last updated in 2002). The search parameters for the minimum number of repetitions, which can be set separately for each motif length, and the maximum number of interruptions used to define a microsatellite are given in an .ini file which needs to be in the same directory as the Perl module.

TROLL (Tandem Repeat Occurrence Locator)

This is a program designed by Castelo *et al.* (2002) to look for perfect microsatellites using a "dictionary approach" based on a slight modification of the *Aho–Corasick* Algorithm (AHO and CORASICK 1975). It requires a list of repeat motifs to be given in an input file, from which a keyword tree is built for finding repeats of the specified motifs. The *Aho–Corasick* Algorithm is then applied to search for the given motifs in the keyword tree and, at every match, a test procedure is called to keep track of the repeats using 'repeat buffers'. The program requires two search parameters (see **figure 2.3**) apart from the input file and the file containing the motifs to search for: maximum motif length (-M) and the 'minimum repeat length' in base pairs (-m, default 20). Based on these, only valid tandem repeats will be saved to the 'repeat buffers', where redundant repeats due to equivalent motifs (ATC=TCA=CAT) are also sorted out by choosing the motif which forms the longest repeat as the valid one.

The program's execution time is expected to have a linear relationship with the length of the query sequence. It is distributed under the GNU General Public License at the Sourceforge page: http://sourceforge.net/project/showfiles.php?group_id=25483 .

```
Usage: troll -M<max pattern length> [-m<min repeat lenght, default 20>] [-c|C]
        <motif file> <chr. file> [<chr. file> ...]
```

Figure 2.3: TROLL usage

Sputnik II

The program Sputnik was originally written by Chris Abajian at the Washington University. The program is written in C and uses recursive searches to find perfect microsatellites with 1 to 5 bp long motif sizes. It also allows for a small degree of imperfection within tandem repeat hits by implementing a scoring system: giving points for matches and subtracting them for mismatches. The original version was modified several

times, first by Morgante *et al.* (2002) at the University of Delaware to speed it up considerably (primarily by limiting unnecessary recursion), to output sequence flanking the repeat (allowing computation of the set of non-redundant repeats), to output the repeat type in terms of a canonical repeat unit (lexicographic equivalent) so that counting can be done regardless of repeat unit phase, or strand, and to report a percent perfection statistic for repeats (the percentage of the repeat length that is composed of an exact repetition of the repeat unit). Subsequently the input and output formats were further modified by La Rota *et al.* (LA ROTA *et al.* 2005); to facilitate the handling of FASTA format and to make the parsing of results easier, respectively. The last version (or Modified Sputnik II) can be obtained from http://wheat.pw.usda.gov/ITMI/EST-SSR/LaRota/ (last viewed on 31.08.09). The parameters for Sputnik II are shown in **figure 1.4**, and, throughout this document, sets of parameters from the Sputnik program will be referred to in the same way they are given into the program (i.e. `-v 1 -u 3 -m 2 -n -6 -s 16 -A -p -L 16 -l -1`).

```
sputnik -umnsfr _fasta_file
where:
-a      show all (even when there is no repeat)
-x      dont bother finding the canonical repeat unit
-u int  max unit length [-v, 5]
-v int  min unit length [1, -u]
-m int  points for a match [1,)
-n int  points for a mis-match (,-1]
-s int  min score [5,)
-j      adjust scores for the first unit cell.
-e int  errors per 100 bases, -ve means ignore [-1,100].
-p      report score as percent perfection.
-f int  fail score (,-1]
-d int  max degrade and still continue, in bases.
-r int  max recursion [0, 100] (0 ==> perfect only)
-R int  max recursion [0, 100], and do recursions only if
        score is at least unit length times score for a match.
        Zero implies only perfect repeats will be found.
-A      set -r automatically (by unit cell), and set -j.
-F int  output this many bases flanking the repeat [0, 1000].
-L int  min length of SSR to report.
-l int  max chars on an output line [-1,).
        -1 means no limit, or one line no matter how long
        0 means dont output the repeat sequence at all
-z      do not collapse unit cell to canonical strand
```

Figure 2.4: Sputnik usage

STRING (Search for Tandem Repeats IN Genomes)

This is a heuristic algorithm initially presented in 1998 by De Fonzo *et al.* (1998), and further modified by the same group of authors (PARISI *et al.* 2003). It is written in standard C language and available from http://www.caspur.it/~castri/STRING/. The program is aimed at finding tandem repeats of any motif size and with a small number of point mutations in comparison to a consensus perfect tandem repeat. Similar to the program TRF, the search

strategy of STRING is divided in two phases: a detection phase and an extension phase. During the first phase a heuristic process is applied to select "interesting zones" which could contain tandem repeats. It also selects the most 'promising' possible consensus words for the extension phase. This selection is based on the search for auto-alignments described in De Fonzo *et al.* (1998). In the second phase, the auto-alignments are grouped into suitable clusters by extending them based on comparisons with the consensus words corresponding to the sequence zone. The program requires five arguments to run: the sequence length, a pointer to the sequence, pointers for two output files, and an integer value representing the threshold score above which tandem repeats should be considered interesting.

mreps

The mreps program was published as a special tool to identify fuzzy repeats in a single run for whole genomic sequences (KOLPAKOV *et al.* 2003), and it is based on an earlier published algorithm (KOLPAKOV and KUCHEROV 2001; KOLPAKOV and KUCHEROV 2003). It is written in C and distributed under General Public Licence (GPL), which means its source is available for modification. The main advantage of this program, according to the authors, is that it can search in a single run for all possible motif sizes, from microsatellites to huge tandem duplications. The algorithm consists of two main parts: the first one is the "upper frame" which collects sequences repeated in tandem through a combinatorial algorithm, and the second is the "lower frame" which applies to them a heuristic treatment to decide which repeats are relevant. The combinatorial method attempts to extend each repeat to the right and left as much as possible, as far as periodicity is respected (see KOLPAKOV and KUCHEROV 2003). For this, it looks for the "maximal run of *k*-mismatch tandem repeats" which still verifies the definition of a tandem repeat. The parameter "*k*" is the maximal number of mismatches allowed between two tandemly repeated copies, an absolute value which is provided by the user. Therefore, the user needs to have *a priori* knowledge of the motif size to specify an appropriate number of imperfections to be expected in the repeats, which restricts the power of the approach (KOLPAKOV *et al.* 2003). During the "lower frame" heuristic treatment the repeat hits are post processed eliminating non-repeating edges, choosing appropriate motifs to report, and filtering out statistically expected tandem repeats. This last filtering is based on computer simulations on pseudo-random DNA sequences by a process not explained in the paper.

The parameters available for fine-tuning the search are given in **figure 2.5.**

```
mreps [ <options> ] { <sequencefile> | -s <sequence> }

The options are :
 -s <string>  : specifies the sequence in command line
 -fasta       : allows DNA sequences in FASTA format

 -res n       : "resolution" (error level)
 -from n      : starting position n
 -to n        : end position n
 -minsize n   : repeats whose size is at least n
 -maxsize n   : repeats whose size is at most n
 -minperiod n : repeats whose period is at least n
 -maxperiod n : repeats whose period is at most n
 -exp x       : repeats whose exponent is at least x
 -allowsmall  : output small repeats that can occur randomly

 -win n       : process by sliding windows of size 2*n overlaping by n
 -xmloutput <file> : outputs to <file> in xml format
 -noprint     : if specified, the repetition sequences will not be output

Example:
mreps -res 3 -exp 3.0 -from 10000 -to 12000 ecolim52.fas
```

Figure 2.5: mreps usage

ptrfinder (Perfect Tandem Repeat Executable)

This is a program written in C by Jack R. Collins based on string comparison techniques. The algorithm has not been published yet, but the program is presented in an application paper by the author (COLLINS *et al.* 2003). As its name suggests, ptrfinder is useful for finding perfect tandem repeats, and the motifs range from 2 to 16 nucleotides in length. Executables for several operating systems including Linux, Solaris, and Mac OS X are available at http://ncisgi.ncifcrf.gov/~collinsj/Tandem_Repeats/downloads/ . According to Collins (2003), future publication of a detailed description of the algorithm is planned. The program's readme file states that the program is based on a simple pattern matching algorithm based on the motif size and the minimum number of tandem repeats required, and thar the result should be an exhaustive search. The available parameters for the program run are shown in **figure 2.6**.

```
usage:./ptrfinder.sgi -seq dna_file -repsiz repeat_size -minrep
min_num_reps > stdout
(Optional switches) -sql sql_basename -ucsc ucsc_out -grid grid_out
(Optional switches) -chrom chromosome/string identifier

 -seq    dna_file in fasta format (REQUIRED)
 -repsiz repeat_size: size of repeating element eg. 2 would find ATATATATAT
(REQUIRED)
         repeat_size: two numbers separated by a comma can indicate a range
of sizes (optional form)
 -minrep min_num_reps: minimum number of times repeated to be reported
(REQUIRED)
 OPTIONAL command parameters: ************************************
 -sql    sql_basename: sql .idx and .tfa output files will be named
         sql_basename: sql_basename_ptr.idx and sql_basename_ptr.tfa
 -ucsc   ucsc_basename: ucsc_basename_ptr.bed in UCSC bed format
 -grid   grid_basename: grid_basename_ptr.feature file for ABCC GRID db
 -chrom  seq identifier: chromosome # or identifier eg. 1, X, Y, etc.
 -chrom  chrom must be specified if either -sql -ucsc or -grid are invoked
```

Figure 2.6: ptrfinder usage

ATRHunter (Approximate Tandem Repeats Hunter)

This is an algorithm specialized for the detection of approximate tandem repeats (ATRs), It has a principle similar to the program TRF in that it divides the search into two phases, a screening phase to detect candidate tandem repeats, followed by a verification phase where the quality of the approximate tandem repeats is verified by sequence alignments with perfect versions of the repeat. The innovation resides in some details in the screening phase, which uses a variable size sliding window (*l-window*, comparable to *k-tuple* in TRF), different similarity measures, as well as a different scoring system (WEXLER *et al.* 2005). This screening phase uses an iterative algorithm, performing one iteration per motif length, and for each iteration the *l-window* size and the quality threshold are selected by the algorithm to adjust the scoring function. Each iteration is called a sliding step, and its size can range between 0.1 and 2 positions, in order to maximize the number of matching l-windows detected. Thanks to this, the algorithm should be able to find more repeats than the program TRF. The verification phase is identical to the one in the program TRF.

The program is available as a command line executable or a graphical interface written in java, both of these for Windows OS. For sequences with less than 2 Mb searches can also be submitted at the ATRHunter web page http://bioinfo.cs.technion.ac.il/ATRHunter/. The search parameters to fine-tune the searches are prompted for by the program in the following order: the alignment parameters or scores for a match, mismatch, indel, and terminal gap, the maximum motif length, the minimum similarity level, and the definition of

ATR to be used, to choose from the three definitions shown in **figure 2.7** as described in the program.

```
1. Similarity level between adjacent copies (default definition)
The definition advocated in our paper considers a sequence to be an ATR with
motif length t if the alignment score between adjacent copies of length t is at
least s*t, where s is the level of similarity chosen by the user.
In addition, the alignment score for matching non-adjacent copies has to be at
least ?*t, where ? is a number determined in the program.
This restriction prevent the dispersal of similarity in an ATR.

2. Similarity level between adjacent copies (also used in the program
TEIRESIAS)
This definition is more permissive than the above definition and was suggested
by Stolovotzky et al. (1999)
It considers a sequence of length c*t to be an ATR with c copies of a motif of
length t if the average alignment score between tandem copies is at least
s*t,where s is the level of similarity chosen by the user.
In addition, the sequence is regarded as an ATR, if there exists a copy the
alignment of which with every other copy is not less than ?*t, where ? is a
number determined in the program. This restriction prevent the dispersal of of
similarity in an ATR but is not as restrictive as its equivalent in the first
definition.

3. Minimum alignment score with a repeating copy (also used in the program TRF)
The definition, which was suggested by Benson and used in the program TRF,
considers an ATR to be a genomic stretch which scores at least s when compared
with a best fitting pattern, where s is the level of similarity chosen by the
user. The score s is fixed and does not vary for different motif lengths.
```

Figure 2.7: Three definitions of ATR to choose from when using ATRHunter

STAR (Search for Tandem Approximate Repeats)

STAR is a program written by Delgrange *et al.* (2004) optimized for finding imperfect tandem repeats or ATRs. The program uses a compression algorithm to find tandem repeats given a specific motif. The motifs to be included in the search are provided in an auxiliary motif file, where the motifs are listed in lexicographic order. The program is suited to search for motifs of any length, but the motif file comes with a list of motifs from 1 to 6 nucleotides long. Based on the motif file, STAR identifies all segments of the sequence that correspond to significant approximate repetitions based on their compression values which are assessed using the "Minimum Description Length criterion" (MDL). This MDL criterion is a formal version of the Occam's Razor principle, where the simplest and shortest hypothesis to explain a phenomenon is usually considered more likely to be the true (DELGRANGE and RIVALS 2004). It evaluates how many mutations are allowed in an ATR based on its perfect counterpart, and this evaluation is independent of motif length.

The nucleotide sequence encoding for a tandem repeat should be easy to compress by keeping only information about the motif and the number of motifs. In the case of an ATR, a

list of mutations occurring within the repeat are also included, increasing in this way the amount of data to encode. For a true ATR, the mutation list is expected to be short, so that it is more economical to encode the motif, the length, and the list of mutations, as to save the whole sequence. This is determined by the program for each possible ATR by computing and analyzing compression curves. The input options for STAR are shown in **figure 2.8**.

```
Usage: star_win32.exe -i SeqFile -m Motif | -M MotifFile [-na -po
PositionOffset
 -help]
   Seqfile: file containing the sequence in Fasta, Genbank or Embl format
   Motif: the motif to search for as a string over alphabet [ACGT]
   MotifFile: a file with one motif per line, each motif is searched
              independently, this option excludes option -m
   -na : option without the output of alignments of tandem repeats;
              default is with alignments
   -po PositionOffset : set a position offset that is added to output
              positions;
```

Figure 2.8: Usage for the program STAR

TRA (Tandem Repeats Analyzer)

This program uses two different algorithms for detecting perfect (exact) and imperfect (inexact) microsatellites respectively. These algorithms are described and applied in Bilgen *et al.* (2004), and in another almost identical version of this publication describing and applying only the algorithm for exact matches eTRA (KARACA *et al.* 2005). TRA and eTRA are distributed as Windows executables and can be downloaded from ftp://ftp.akdeniz.edu.tr/Araclar/TRA/. The exact module e-TRA is faster and has more parameters to specify the search.

The graphical interface for the TRA executable is shown in **figures 2.9** and **2.10**. As can be deduced from these figures, the program TRA is specialized for the development of microsatellite markers, allowing input of queries in Genbank and EST format, and performing the corresponding classification of microsatellites per sequence region. It can also output flanking sequences for the design of primers. According to the authors, TRA offers the ability to identify compound repeats, and to inform researchers about distribution of repeats in organs, tissues, cell types, and developmental stages.

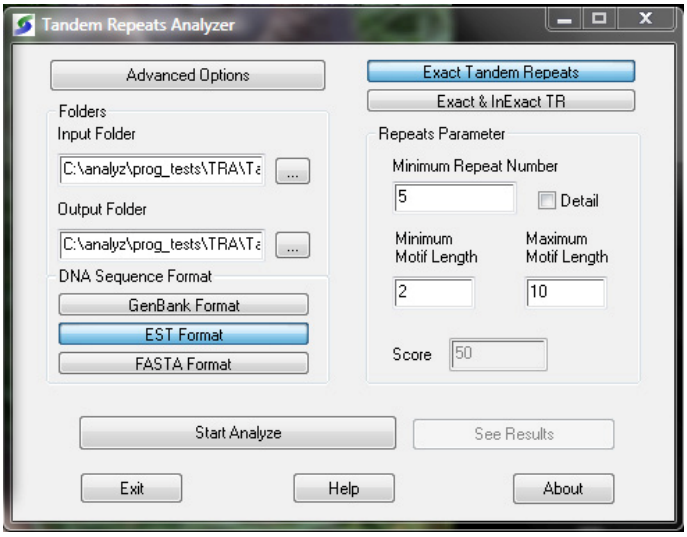

Figure 2.9: Graphical interface of the TRA program showing the available options. The "Details" option allows specifying minimum number of repeat values independently for each motif, but it is only available for the exact tandem repeat search.

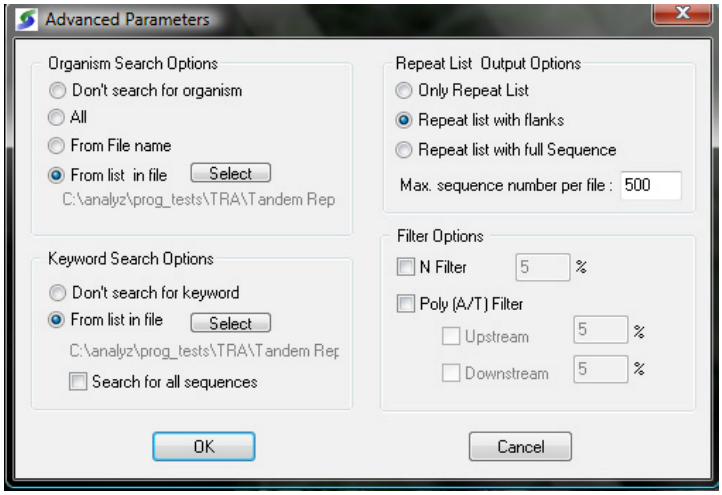

Figure 2.10: Pop-up window shown when selecting the "Advanced options" of the main window shown in figure 2.9.

Msatfinder

This is another program written in Perl for which the algorithm was not officially explained in a scientific publication. It is distributed by the authors at http://www.genomics.ceh.ac.uk/msatfinder/ , where an online interface, a user manual, and links to other tools are also available. The program has a General Public Licence (GPL), and it

has the Bioperl and EMBOSS packages as dependencies. As the name of the program indicates, it is specific to find microsatellites (1 to 6 bp motifs) in nucleic acid or protein sequences. It can also detect longer motifs and no specific upper limit is stated. However, minimum size thresholds have to be specified manually and individually for each motif. The program only reports hits with perfect tandem repetitions.

The output of Msatfinder consists of multiple files either summarizing or detailing different aspects of the results. For the output, several format options are offered to facilitate post-processing: details about the repeats, counts, feature tables for input into the visualization program Artemis, the microsatellite sequences in FASTA format, and possible primers for each microsatellite. Further details on motif and motif size counts are also automatically provided.

The parameters for the search need to be specified beforehand in an auxiliary file: msatfinder.rc, which has to be in the same directory as the program. The parameters are summarized in **figure 2.11**.

- debug - if set to 1, will print extra debugging information. Set to 2 for even more.
- flank_size - the amount of sequence either side of the microsatellite that will be extracted and saved to the microsatellite FASTA file.
- mine_dir,repeat_dir,tab_dir,bigtab_dir,fasta_dir,prime_dir,align_dir,cont_dir - several variables that set the name of the subdirectories that will be created when the script is run.
- run_eprimer - set to 1 if you want to determine whether a primer can be made for each repeat.
- eprimer_args - the eprimer man page has more information on what to put here, if you are dissatisfied with the default (pick PCR primers). Please note that the "-task 0" option works with EMBOSS 2.8.0. If you have 2.9.0 then you should use "-primers" instead.
- eprimer - full path to the eprimer3 binary.
- primer3core - the full path to the primer3_core binary.
- override - turns off the following variables. It's easier than editing lots of lines in the config file.
    - artemis.
    - mine.
    - fastafile.
    - sumswitch.
    - screendump.
    - run_eprimer.
- motif_threshold - this is particularly important, as it defines the thresholds **equal to or above** which microsatellites will be detected, and which types will be detected. The types may be set to any length, and the lowest the thresholds can be set is 1, which will find every single base, pair of bases, triplet &c. It will take a long time to run if thresholds are set that low and the "regex" engine will not operate on such a small threshold. By default, mono-hexa will be searched for. Please refer to setting thresholds and motif types (below) for more information.

Figure 2.11: Extensive set of parameters to optimize searches with Msatfinder

- artemis - turns on the Artemis feature tables.
- mine - turns on MINE summary files. These are equivalent to the "repeats" output file in the data they contain.
- fastafile - turns on whether or not a FASTA format file containing the sequence information for each microsatellite found will be generated.
- taxon_information - two of the fields in the repeats and sequence files are "specific_taxon" and "generic_taxon". See here.
- remote_link - used to put a hyperlink into MINE files for looking at the original genomes.
- sumswitch - determines whether or not the "repeats" output file will be created. This contains a large amount of information about each microsatellite and its genomic context, and can become rather large. However, it is very useful for importing into a database.
- screendump - prints out verbose information to the screen whilst running if set to 1.

Figure 2.11: Extensive set of parameters to optimize searches with Msatfinder (continued)

Msatfinder is part of the Msatminer project (http://www.genomics.ceh.ac.uk/msatminer/), which is a collection of perl scripts designed for the investigation and characterization of microsatellite markers. The other perl scripts allow the post-processing, handling, and analysis of Msatfinder data.

IMEx (Imperfect Microsatellite Extractor)

This is a program written in C which was published in March 2007 (MUDUNURI and NAGARAJARAM 2007). Compiled binaries for linux as well as a web server to run the program online are available at the IMEx Webpage: http://203.197.254.154/IMEX/. The program is specific for microsatellites with motifs from 1 to 6 bp in length, with perfect as well as imperfect repeat units. The program is divided in two phases, the first being a detection phase based on a "simple-string matching algorithm" which uses a sliding window approach to screen DNA sequences and find "nucleation sites". The second phase extends the nucleation sites on both sides in steps as long as the imperfection characteristics specified by the user are met. Only one indel per repetition is allowed in a microsatellite, and the minimum number of repetitions, maximum number of point mutations, and the maximum percentage of imperfections (point mutations and indels) allowed per hit can be specified individually for each motif size (MUDUNURI and NAGARAJARAM 2007). Additionally, the program can include microsatellite flanking sequences in the output, and in combination with primer3 (a program developed by ROZEN and SKALETSKY 1998) it can be used to design primers for the reported microsatellites. The search options available for this program are shown in **figure 2.12**.

```
ENTER THE 'k' VALUES (Imperfection limit/repeat unit):
Mono [0-1]:
Di [0-2]:
Tri [0-3]:
Tetra [0-4]:
Penta [0-5]:
Hexa [0-6]:

ENTER THE 'p' VALUES (Imperfection percentage):
Mono [0-90]:
Di [0-90]:
Tri [0-90]:
Tetra [0-90]:
Penta [0-90]:
Hexa [0-90]:

ENTER THE 'n' VALUES (number of repeat units/tract):
Mono:
Di:
Tri:
Tetra:
Penta:
Hexa:
```

Figure 2.12: Example of the input options of the program IMEx after pronmping it from the command line. The program sequentially prompts for each of the search parameters shown here.

SciRoKo (SSR Classification and Investigation by Robert Kofler)

This is a program written in C published in mid 2007 (KOFLER *et al.* 2007b). The underlying algorithm is very efficient, allowing searches of complete chromosomes to be run on a Desktop PC, and it has an easy to use graphical user interface. The program can search for perfect and imperfect (mismatched) microsatellites, offering four search modes, two for perfect and two for imperfect microsatellites. The main difference among the modes is the way the penalty scores are calculated based on the parameters available. Additionally, it also incorporates a search mode based on the MISA algorithm. The program was initially only available in GUI version, the parameters for which are shown in **figure 2.13**. On request, the author made available a command line version called SciRoKoCo, which has a reduced set of parameters compared to the GUI (**figure 2.14**), but it offers the advantage of being fully automatable.

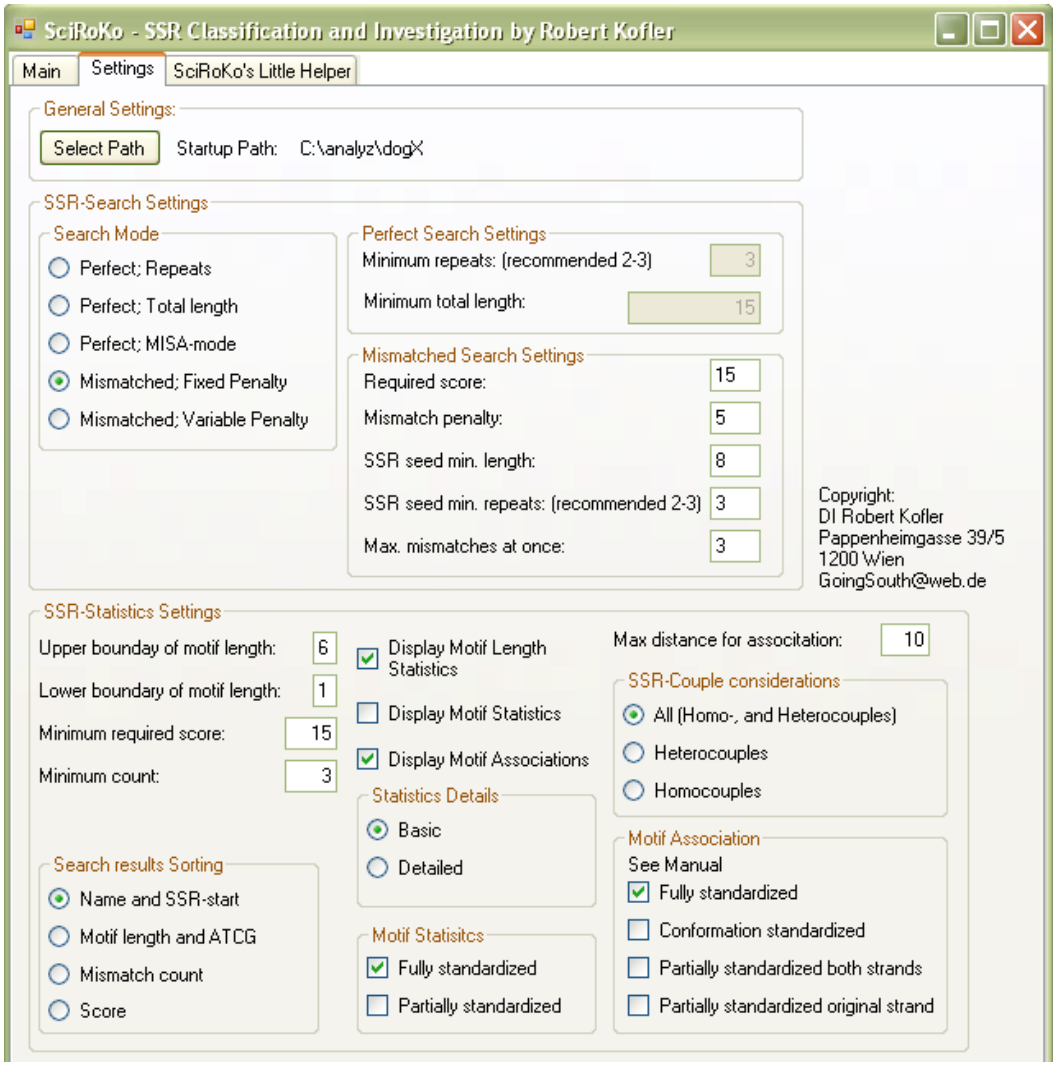

Figure 2.13: Range of parameters offered by SciRoKo to fine tune microsatellite searches

The original publication of the program as an "Application note" in the Journal Bioinformatics (KOFLER *et al.* 2007b) does not give information about the program algorithm, only very general statements about the program's usability. However, a manual for SciRoKo is available on the web page (http://kofler.or.at/bioinformatics/index.html KOFLER *et al.* 2007a), extending the information on the algorithm's functioning.

```
iris@Thinky /cygdrive/c/analyz/prog_tests/SciRoKoCo
$ ./SciRoKoCo.exe
SciRoKoCo; The SciRoKo Console for Microsatellite Search
Example call: SciRoKoCo -i input1.fa -i input2.fa
Example call: SciRoKoCo -i input.fasta -mode mmfp -s 15 -seedl 9 -seedr 4 -p 5 -o result.td

Obligatory parameters:
-i            :input file(s); (multiple) fasta files; obligatory parameter
              :several files may be specified

Optional parameters:
-o            :output file; optional paramter; default: result.sciRo
                 Attention; the extension of the output file has to be either: '.sciRo' or '.td'
                 '.sciRo' is the recommendet output and '.td' is in tabdelimited for easy parsing
-mode         :SSR-search mode; optional parameter; default: mmfp
               possible modes are: pl, pr, misa, mmfp, mmvp
               pl - perfect length: search for perfect microsatellites specify the minimum length
               pr - perfect repeats: search for perfect microsatellites specify the minimum repeats
               misa - misa mode: search for perfect microsatellites specify the minimum repeats for
                 each microsatellite type individually, eg. 14-7-5-4-4-4
               mmfp - mismatched fixed penalty: search for imperfect microsatellites using a
                  minimum score and a fixed mismatch penalty (default mode)
               mmvp - mismatched variable penalty: search for imperfect microsatellites using a
                  minimum score and a variable mismatch penalty; penalty = motifLength x specifiedValue


Parameters for pl-mode (perfect length):
-l            :minimum length of a microsatellite in bp; optional parameter; default 15
-r            :minimum repeats of a microsatellite; optional parameter; default 3
               set this parameter to zero (0) if only the length should be considered

Parameters for pr-mode (perfect repeats):
-r            :minimum repeats of a microsatellite; optional parameter; default 5

Parameters for misa-mode):
-m            :minimum repeats for each microsatellite type; optional parameter; default 14-7-5-4-4-4
               e.g: at least 14 repeats for a mononucleotide microsatellite
                    at least 5 repeats for a trinucleotide microsatellite and so on

Parameters for mmfp-mode (mismatched fixed penalty):
-s            :score; optional parameter; default 15
-p            :mismatch penalty; optional parameter; default 5
-seedl        :minimum length of a SSR-seed; optional parameter; default 8
-seedr        :minimum repeats of a SSR-seed; optional parameter; default 3
-mmao         :maximum mismatches at once; optional parameter; default 3
               this value refers to the depth of the recursion, cpu time increases with this value

Parameters for mmvp-mode (mismatched variable penalty):
-s            :score; optional parameter; default 15
-p            :variable mismatch penalty; optional parameter; default 1
               the mismatch penalty is calculated = variablePenalty x SSR-motif length
-seedl        :minimum length of a SSR-seed; optional parameter; default 8
-seedr        :minimum repeats of a SSR-seed; optional parameter; default 3
-mmao         :maximum mismatches at once; optional parameter; default 3
```

Figure 2.14: Search parameter options for SciRoKoCo (command like version of SciRoKo)

Tandem

This is an algorithm written by Domanic and Preparata (2007). Its repeat finding strategy is very similar to the one used in TRF, having a detection phase where candidate tandem repeats are identified, and a verification phase where these candidates are verified by sequence alignments. The verification phase is similar to the one from TRF, and therefore also to the one used in ATRHunter, and the main innovation is introduced in the detection phase, where only immediately preceding occurrences of short sequence strings (windows) at every position are compared. The authors claim that this procedure is as effective as the one in TRF, which looks at all previous windows, while the running time is dramatically improved. This is expected to increase the efficiency of the algorithm without affecting its detection accuracy. The stochastic process of the formation of approximate tandem repeats

is also similar to the one in TRF, and therefore, as can be seen in figure 2.15, the parameters required to run tandem are the same as for TRF, except for the mu+ parameter.

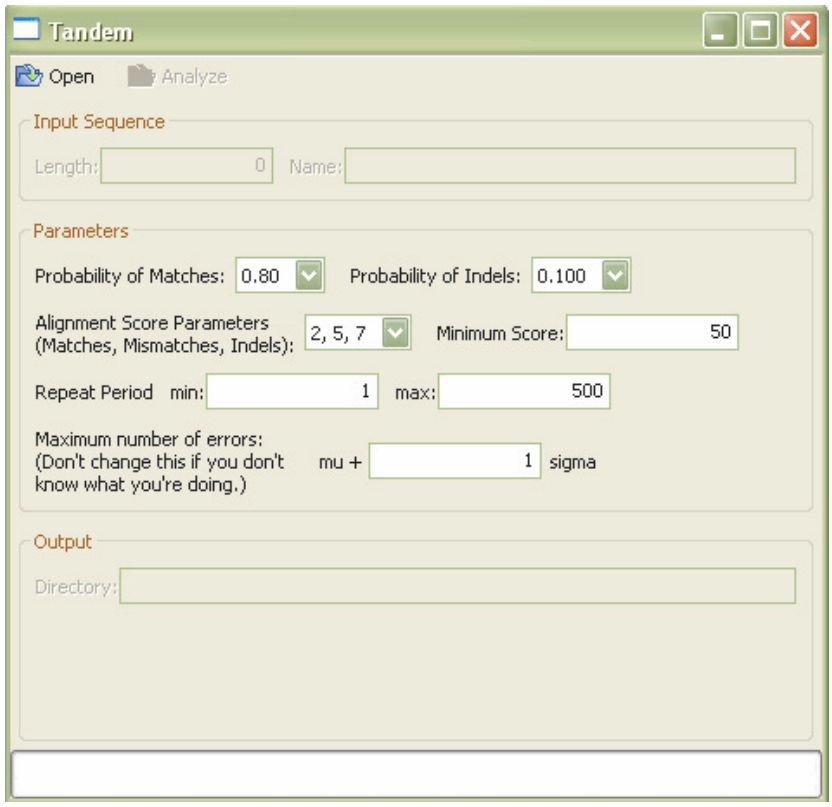

Figure 2.15: Graphical interface and parameters offered by tandem

TRED (Tandem Repeat Software over the Edit Distance)

This recently published algorithm for detecting approximate tandem repeats in genomic sequences (SOKOL *et al.* 2007) is an extension to the algorithm from Landau (1998). The authors define tandem repeats using a model of evolvable tandem repeats, which assumes that each repeat unit, from left to right, is derived from the previous copy through zero or more mutations. Thus, each copy of the repeat should be similar to its predecessor and successor copy.

The program's algorithm implements search methods applied by algorithms based on Hamming distance definitions, but uses an edit distance definition. The searches are supposed to be deterministic because the definition used is rigorous and the program looks for all repeats that match the definition. The main problem here is the excess of redundant hits that would be produced, and the additional time to process these. This is resolved by

filtering out the redundant hits before they are reported using within- and between-iteration filtering procedures. The computation time of the program is further reduced by using partial edit distance matrices and by the overall reduction of iterations based on the algorithm by Main and Lorenz (MAIN and LORENTZ 1984).

Phobos

This program was developed by Christoph Mayer at the RHUR University of Bochum. There is as yet no scientific publication describing Phobos, but a complete user manual which includes a brief description of the algorithmic strategy is available (MAYER 2007). According to its web page (http://www.ruhr-uni-bochum.de/spezzoo/cm/cm_phobos.htm), the program is still under development, and it has already been used in three genome projects. These genome projects are, however, not identified nor cited. The latest version of Phobos, 3.3.2, was released in August 2008.

Phobos is based on an "exact algorithm" (non-probabilistic), and it can search for tandem repeats with a motif of 1 to 5000 nt in length. It can also handle imperfections, both indels and substitutions, as well as detect repeats with several different motifs. The algorithm is based on a scoring system, where the score is based on local alignments of the putative repeat with a perfect tandem repeat of its motif. Alignments are performed in both directions, and the alignment with the best score is used to decide if the repeat should be extended or not. Therefore, the search is independent of search direction.

Phobos has a long set of input options (see **figure 2.16**), including different search modes, search arguments, and output options. Among the output options, the program can report several output formats, alignments of the repeats, and flanking sequences.

```
    phobos_cl.exe  [-M <exact|extendExact|imperfect>] [-g <int>] [-m <int>]
                   [-s <int>] [--minScore_a <int>] [--minScore_b <float>]
                   [-l <int>] [--minLength_a <int>] [--minLength_b <float>]
                   [-U <int>] [-u <int>] [-r <int>] [--lastSeq <int>]
                   [--firstSeq <int>] [-D] [-P <float>] [-f <int>] [--maskX]
                   [--outputFormat <int>] [--printRepeatSeqMode <int>]
                   [--NPerfectionMode <int>] [--reportUnit <int>]
                   [--NsAsMissense] [-N <int>] [--] [-v] [-h] <Input and
                   output filenames> ...
Where:
-M <exact|extendExact|imperfect>,  --searchMode <exact|extendExact
      |imperfect>
     (value required)  Phobos provides three different search modes. exact:
     Search for exact repeats, only. extendExact: Searches for exact
     repeats and extends them by inserting mismatches and indels.
     imperfect: Searches directly for imperfect repeats.
```

Figure 2.16: Phobos usage

```
-g <int>,  --indelScore <int>
     (value required)  Score for indels - must be negative. Default: -6.
     Match score is fixed to one.

 -m <int>,  --mismatchScore <int>
     (value required)  Score for mismatch - must be negative. Default: -6.
     Match score is fixed to one.


--minScore_a <int>
     (value required)  The minimum score of a repeat is determined with:
     maximum( minScore, minScore_a + minScore_b*(unit-length) ). Default
     value of minScore_a: 0

 --minScore_b <float>
     (value required)  The minimum score of a repeat is determined with:
     maximum( minScore, minScore_a + minScore_b*(unit-length) ). Default
     value of minScore_b: 0

 -l <int>,  --minLength <int>
     (value required)  The minimum length of a repeat is determined with:
     maximum( minLength, minLength_a + minLength_b*(unit-length) ). Default
     value of minLength: 1

 --minLength_a <int>
     (value required)  The minimum length of a repeat is determined with:
     maximum( minLength, minLength_a + minLength_b*(unit-length) ). Default
     value of minLength_a: 0

 --minLength_b <float>
     (value required)  The minimum length of a repeat is determined with:
     maximum( minLength, minLength_a + minLength_b*(unit-length) ). Default
     value of minLength_b: 0

 -U <int>,  --maxUnitLen <int>
     (value required)  Maximum unit length. Default: 5

 -u <int>,  --minUnitLen <int>
     (value required)  Minimum unit length. Default: 2

 -r <int>,  --recursion <int>
     (value required)  The recursion depth used in the search. Values in
     the range 3 to 7 are recommended. A value of 0 implies a search for
     perfect repeats only.

 --lastSeq <int>
     (value required)  Number of last sequence to be processed in this run.

 --firstSeq <int>
     (value required)  Number of first sequence to be processed in this run.

 -D,  --dontRemoveMostlyOverlapping
     Phobos sometimes finds repeats that partially or completely overlap
     with other repeats, e.g. if alignments with alternative repeat
     patterns exist at the same locus in a sequence. The default is to
     remove one of any two mostly overlapping repeats in favour of that
     repeat with the highest score. With this option, Phobos reports also
     repeats that mostly overlap with higher scoring repeats.

 -P <float>,  --minPerfection <float>
     (value required)  Minimum perfection of satellites. Satellites below
     this value are broken up or reduced in length if this yields a
     satellite above the minimum perfection. If N's are treated as mismatch
     when computing the perfection, this value also has an influence when
     searching for exact repeats.

 -f <int>,  --flanking <int>
     (value required)  If the satellite sequence is printed, this is the
     number of flanking nucleotides to be printed to the left and right of
     it. Default: 0
```

Figure 2.16: Phobos usage (continued)

```
 --maskX
     Writes the sequences to a file with the extension ".masked" in which
     repeats are masked.
--outputFormat <int>
     (value required)  Phobos provides different output formats for
     printing the repeat information. 0: Phobos output format, 1: extended
     Phobos output format

   --printRepeatSeqMode <int>
     (value required)  Phobos provides different modes to print the repeat
     sequence along with its information. 0: don't print sequence, 1: print
     sequence, 2: print alignment.

   --NPerfectionMode <int>
     (value required)  Phobos provides different modes to treat N's when
     computing the perfection of a repeat. 0: asMismatch, 1: asNeutral, 2:
     asMatch. Default: 0

   --reportUnit <int>
     (value required)  Repeat units can be reported in three different
     modes. 0: asIs, 1: Alphabetical normal form, 2: Alphabetical normal
     form also considering the reverse complement. Default: 2
   --NsAsMissense
     Treat N's as missense. Default: Treat N's as neutral with score 0.

   -N <int>,  --succN <int>
     (value required)  The maximum number of successive N's allowed in a
     satellite. Default: 2.
   --,  --ignore_rest
     Ignores the rest of the labeled arguments following this flag.
   -v,  --version
     Displays version information and exits.
   -h,  --help
```

Figure 2.16: Phobos usage (continued)

2.1.3 Microsatellite databases

Several websites provide access to microsatellite databases for a variety of species. These have been designed, often by the authors of the repeat finding programs themselves, to facilitate the development of microsatellite markers in sequenced genomes (**Table.2.2**). They usually provide easy to use interfaces to retrieve and manipulate microsatellite sequences. The downfall of these databases is that, although the authors performed all the search runs and have a wealth of information at hand, none of the databases cited in **table 2.2**, except for InSatDb (ARCHAK *et al.* 2007), offers statistical or quantitative information for the contents of the database. In other words, it is possible to retrieve single microsatellites and to design primers from them, but general information about microsatellite distribution, abundance, motif preference, or imperfection content in microsatellites of the featured genome can usually not be retrieved. Therefore, these databases are effectively of no use for comparative genomics studies.

Table 2.2: Main microsatellite databases publically available (not an exhaustive list)

Browser	Link publication	Repeat finding program used	Program parameters	Information provided	Includes imperfect microsatellites	Observations
UCSC Genome Browser	http://www.genome.ucsc.edu	TRF (BENSON 1999)	Provided by request 2 7 7 80 10 50 2000 -m Not customizable	Eukaryotic microsatellite and minisatellite positions and sequence	Yes	Datasets have not been filtered for redundancy
TRDB Tandem Repeat Database	http://cagt.bu.edu/page/TRDB_about	TRF (BENSON 1999) Mreps (KOLPAKOV et al. 2003)	—	—	Yes	It is currently (May 2009) an experimental system which is undergoing frequent changes
The Microorganisms tandem repeat database	http://minisatellites.u-psud.fr/GPMS/	TRF (BENSON 1999)	Customizable	Minisatellites and microsatellites for Bacteria, Archaea and Viruses	Yes	Used to characterize minisatellites in Yersinia pestis and Bacillus anthracis (LE FLECHE et al. 2001)
TRbase	http://trbase.exeter.ac.uk/advtr.html	TRF (BENSON 1999)	2 5 5 80 10 45 2 7 7 80 10 45 Customizable	Tandem repeats with 1 to 2000 nt motifs for the human genome build 34	Yes	The cut-off value of 45 (last parameter) is too high for microsatellites.
InSatDb	http://sunserver.cdfd.org.in:9999/PHP/INSATDB/home.php ARCHAK et al. 2007	TRF (BENSON 1999)	2 3 5 80 10 30 2 5 7 80 10 30	Microsatellite sequences of five genomes: fruitfly, honeybee, malarial mosquito, red-flour beetle and silkworm. It includes information on genomic location and imperfection. Microsatellites are grouped into families based on conservation of flanking regions	Yes	Its purpose is comparative genomics
ABCC GRID Database Short Tandem Repeats	http://grid.abcc.ncifcrf.gov/str.php COLLINS et al. 2003	Perfect Tandem Repeat Finder (ptrfinder)	Customizable	Microsatellite counts in several Eukaryotic genomes	No	—
MICdb2.0 from MICAS Microsatellite Analysis Server	http://sunserver.cdfd.org.in:8080/MIC/index.html SREENU et al. 2003	W-SSRF Web-Simple sequence repeat finder	Customizable	Microsatellite motif counts for Bacteria, Archaea, and Viruses	No	—

Table 2.2: Main microsatellite databases publically available (continued)

Browser	Link publication	Repeat finding program used	Program parameters	Information provided	Includes imperfect microsatellites	Observations
EuMicroSatdb (**Eu**karyotic **Micro**sat**e**llite database)	http://ipu.ac.in/usbt/EuMicroSatdb.htm	MISA - MIcroSAtellite identification tool	Customizable	Positions and flanking region sequence for microsatellites in several Eukaryotic genomes	No, only perfect and compound perfect	The information for microsatellites with specified motif and length can be retrieved through a series of detailed menus.
Small Genomes Microsatellite database	http://www.genomics.ceh.ac.uk/cgi-bin/sgmd/index.cgi	Msatfinder	Customizable	Minisatellites and microsatellites for Bacteria, organelles, and Viruses	No	—
SilkSatDb	http://sunserver.cdfd.org.in:9999/PHP/SILKSAT/index.php PRASAD et al. 2005	Simple Sequence Repeat Finder (SSRF) www.cdfd.org.in/micas	Customizable	Microsatellite data from ESTs and whole genome sequences of the silkworm Bombyx mori. It is the only database offering info about distribution and abundance of microsatellites, polymorphism and mapping	No	They used the same algorithm used in the MICAS database. This algorithm was said to be in preparation to be published in 2003, but it was never published.
CMD Cotton Microsatellite Database	http://www.cottonmarker.org/	No specific program, experimental data.	Customizable	Sequences, primers, map data, homology	No	It contains a standardized panel of 8,915 publicly available SSRs for cotton
VNTRDB Variable Number Tandem Repeat Database	http://vntr.csie.ntu.edu.tw/.	Variable number tandem repeat-PCR (VNTR-PCR)	Not customizable	Bacterial potentially polymorphic tandem repeats with inter-genus and intra-genus conservation	Yes	Contains microsatellites and minisatellites
Tandem Repeat Database (TRED)	http://www.sci.brooklyn.cuny.edu/~sokol/tandem/?view=reader&limit=0,25	Tandem Repeat Software (SOKOL et al. 2007)	Not available	—	Yes	—

2.2 Methodology

2.2.1 Programs reviewed and tested

I reviewed the current literature looking for software tools with the capacity to look for microsatellites on whole eukaryotic chromosomes. The list of programs reviewed here is presented in **table 2.3**.

Table 2.3: Initial list of programs reviewed and/or tested in the present study

Year	Program	Language	Webpage	Publication
1997	**Tandyman**	Perl	http://hemisphere.lanl.gov/tandyman/cgi-bin/tandyman.cgi	*NP (LEACH and CLELAND 1997)
1999	Tandem Repeat Finder (**TRF**)	C	http://tandem.bu.edu/trf/trf.html	(BENSON 1999)
2000	**SSR screener**	C	ftp://ftp.technion.ac.il/pub/supported/biotech/	*NP (GUR-ARIE *et al.* 2000)
2001	**SSRIT** Simple Sequence Repeat Identification Tool	Perl	http://www.gramene.org/db/searches/ssrtool	*NP (TEMNYKH *et al.* 2001)
~2002	MIcroSAtellite identification tool (**MISA**)	Perl	http://pgrc.ipk-gatersleben.de/misa/	*NP Author: Thomas Thiel (THIEL *et al.* 2003)
2002	**ComplexTR**	__	http://malawimonas.bcm.umontreal.ca:8091/anabench/Anabench-Jsp/Applications/ListApplications.jsp	(HAUTH and JOSEPH 2002)
2002	Tandem Repeat Occurrence Locator (**TROLL**)	__	http://finder.sourceforge.net/ http://al.jalix.org/FORRepeats/ (the link seems to be broken)	(CASTELO *et al.* 2002)
2003	**Sputnik II**	C	http://wheat.pw.usda.gov/ITMI/EST-SSR/LaRota/	*NP (LA ROTA *et al.* 2005)
2003	Search for Tandem Repeats IN Genomes (**STRING**)	C, java	http://bioinf.dms.med.uniroma1.it/JSTRING/	(PARISI *et al.* 2003)
2003	**Poly**	Python	http://www.bioinformatics.org/poly/wiki/	(BIZZARO and MARX 2003)
2003	**mreps**	C	http://bioinfo.lifl.fr/mreps/	(KOLPAKOV *et al.* 2003)
2003	**SSRfinder**	Perl	http://www.maizemap.org/bioinformatics/SSRFINDER/	*NP Author: Steven Schroeder, SchroederSG@missouri.edu

Year	Program	Language	Webpage	Publication
2003	Perfect Tandem Repeat Executable (**ptrfinder**)	C	http://ncisgi.ncifcrf.gov/~collinsj/Tandem_Repeats/downloads/	*NP Author: Jack R. Collins (COLLINS *et al.* 2003)
2004	Approximate Tandem Repeats hunter (**ATRhunter**)	java	http://bioinfo.cs.technion.ac.il/atrhunter/ATRHunter.htm	(WEXLER *et al.* 2005)
2004	Search for Tandem Approximate Repeats (**STAR**)	__	http://atgc.lirmm.fr/star/	(DELGRANGE and RIVALS 2004)
2004	Tandem Repeat Analyzer (**TRA** and **E-TRA**)	C++	__	Described very briefly in (BILGEN *et al.* 2004), (KARACA *et al.* 2005)
2005	**Msatfinder/Msatminer**	Perl	http://www.genomics.ceh.ac.uk/msatfinder/	*NP (THURSTON and FIELD 2005)
2006	**SSRscanner**	Perl	No web page	(ANWAR and KHAN 2006)
2006	**Phobos**	C++	http://www.ruhr-uni-bochum.de/spezzoo/cm/cm_phobos.htm	*NP But a complete user manual is available, which also explains how the program works (MAYER 2007)
2006	**FireμSat:**	__	http://www.dna-algo.co.za/	(DE RIDDER *et al.* 2006)
2007	Imperfect Microsatellite Extractor (**IMEx**)	C	http://bioinfo.lifl.fr/mreps/	(MUDUNURI and NAGARAJARAM 2007)
2007	Tandem Repeat Software (**TRED**)	C++	http://www.sci.brooklyn.cuny.edu/~sokol/tandem/	(SOKOL *et al.* 2007)
2007	**SciRoKo**	C	http://www.kofler.or.at/Bioinformatics/SciRoKo/index.html	(KOFLER *et al.* 2007b)
2007	**tandem**	__	http://www.cs.brown.edu/people/domanic/tandem/	(DOMANIC and PREPARATA 2007)

*NP : No scientific journal publication was available describing the algorithm. Therefore I cite directly the authors and/or the application paper where the program was first used.

2.2.2 Computer systems

The programs were tested using a personal computer with an Intel PentiumIV 3.2 GHz processor with 3.1 GB in RAM. The operating systems used were Windows XP and Linux Fedora core 4.

2.2.3 Testing and selection process

The program selection process focused on the following characteristics for each program:

- Capacity to identify microsatellites without *a priori* knowledge of the composition of the repeat (motif type, nucleotide composition, number of imperfections).
- Capacity to search for short tandem repeat sequences within large DNA sequences of the scale of eukaryotic chromosomes (~30 to 700 Mb in length).
- Good speed and scalability with respect to the number and length of input sequences.
- Capacity to handle insertion/deletion and substitution imperfections in order to find start and end positions for imperfect (interrupted by point mutations and indels) and complex microsatellites (with more than one motif). Additionally, the possibility to modify penalties for mismatches and indels (insertion/deletions) was desirable since the thresholds for minimum microsatellite length and maximum imperfection are poorly defined characteristics in microsatellites (see Chapters III and IV).
- Capacity to handle both upper and lower case letters, and to ignore any other character, in order to deal properly with hard- and soft-masked sequences, which are commonplace in first drafts of genomes.
- Proneness for automation through bash scripts, so that whole genomes can be run sequentially. This is important due to the exponential growth of genomic sequence data in databases; microsatellite search analyses and other sequence annotation tasks need to be repeatable in an efficient and error-free way.

Usually, not all required information about the characteristics sought-after in the programs was included in the respective publications. Therefore, I divided the selection process into several steps. I first tested the overall functionality of each program, and then the suitability of each program for the tasks I planned to fulfil. The steps taken for the selection and testing process are described below:

Pre-selection.- Each program was run first with default parameters on a set of test sequences consisting on small chromosomes or partial sequences from prokaryotes and eukaryotes, with and without gaps (Ns), and covering a range of CG compositions (see **table 2.4**). The aim of the preselection was to check in first instance if the program versions

obtained worked properly on the systems available, and to assess the aforementioned characteristics:

- Capacity to process large DNA sequences.
- Capacity to handle other IUPAC characters besides the basic four nucleotides.
- Capacity to handle imperfections within tandem repeats: insertions, deletions, substitutions, or combinations of different motifs.
- Potential for automation, for which the input modality and program dependencies are important, as well as the output format options.

The programs which performed satisfactorily during the preselection process were included in the subsequent benchmarking process.

Table 2.4: Test sequences

Sequence name	Species	Size [nucl]	CG [%]	N [%]	Description
NC_003997.fa	*Bacillus antracis* str. Ames	5227293	35.38	0	complete genome
danio.fa	*Danio rerio* (Chr 1: 1-13442)	13442	31.47	12.06	random fragment containing several gaps
zubeca.fa	*Canis lupus familiaris* (Chr6:10081392-10082792)	1400	42.78	0	small clone sequence from the dog genome containing the microsatellite ZuBeCa16, accession number AC093712
plas1.fa	*Plasmodium falciparum* (Chr1	643292	20.55	0	complete chromosome
yeast1.fa	*Saccharomyces cerevisiae* (Chr I)	230208	39.75	0	complete chromosome, NC_004325.fna
hum22.fa	*Homo sapiens*	49691432	47.98	29.86	complete chromosome
custom.fa	---	995	36.98	0	a custom sequence with motifs and imperfect repeats that are usually difficult to identify and/or report properly

Benchmarking runs on selected programs.- This was the process of evaluation of program's capabilities based on the observation of the effects of search parameter changes on the behaviour and output of the different programs. The search parameters for each program were first screened to find 'synonymous parameters' (parameters which modify the same or similar characteristics of the search in different programs). For each program, all parameters that directly or indirectly affect the tandem repeat search were tabulated for comparison (**Table A1** in the appendix section). Similar parameters can have different

effects depending on the algorithm, and also different ranges of action. I therefore constructed microsatellite number and coverage distributions for each program to characterize the range of different results that each program could produce. The data for the distributions was obtained from serial runs for each program with all possible search parameter value combinations (for all parameters which were essential for defining the microsatellite search), varying one parameter at a time. The results were summarized in terms of numbers and nucleotide coverage of microsatellites. The nucleotide coverage is simply defined by the total amount of nucleotides covered by the microsatellites in the search results.

The parameters with stronger effects on the number and length distribution of microsatellites obtained were selected for more detailed benchmarking, also by using microsatellite number and coverage distributions, and for comparison among programs.

Comparison among programs.- Large-scale visual comparisons among programs were perfomed using the microsatellite number and coverage distributions constructed during the benchmarking process. The parameter combinations for which different program datasets produced similar distribution curves indicated the parameter value ranges with which different programs would be searching for similar microsatellite characteristics. In this way I could discriminate the the ranges of parameter values to use for more detailed comparisons among programs.

To compare the search results among programs, and to determine the completeness of these results, I obtained exhaustive perfect repeat datasets from all test sequences except the human chromosome 22 by using a custom perfect repeat finder IrSa. This program was developed for counting tandem repeats of specific motifs in Chapter IV, and is therefore explained in detail in the mentioned chapter. The comparison of search results between the tested programs and the IrSa reference datasets were carried out using the interval manipulation tools at the Galaxy webpage (http://g2.bx.psu.edu, GIARDINE *et al.* 2005). A closer observation and comparison of results from the shorter test sequences was performed with the aid of the sequence visualization tool Bioedit ver 7.0.5.3 (HALL 1999). During the visual comparisons, the ability of each program, and of each parameter combination within these, to extend through imperfections was tabulated as categorical variables on Excel files

2.3 Results and Discussion

The majority of programs reviewed here have some feature or problem in their algorithm which makes them unsuitable for unbiased whole genome microsatellite identification. These features were probably important, or otherwise not a hindrance, for the specific purposes that the programs were originally created for. However, this situation shows the importance of getting a relatively deep understanding of a program's functioning before using it. Here I present the preliminary testing of 20 tandem repeat finding programs, and the further comparison among two of these. Three programs, FireµSat (DE RIDDER *et al.* 2006), TRED (SOKOL *et al.* 2007), and Phobos (MAYER 2007) only became available or were still under development during the final stages of my research, and were therefore not tested. The outstanding characteristics and possible drawbacks of these new programs are described in **table 2.5**. Two additional programs from the list presented in **table 2.3** could not be tested because it was not possible to retrieve the software from the authors: ComplexTR (HAUTH and JOSEPH 2002) and SSRscanner (ANWAR and KHAN 2006). The algorithm and proposed classification of ATRs from Hauth and Joseph (2002) would still be worth exploring in the future.

Table 2.5: Programs that were not tested because they were still under development at the time of writing

Program	Publication	Outstanding characteristics	Possible drawbacks
FireµSat	(DE RIDDER *et al.* 2006)	—	—
Phobos	*NP But a complete user manual is available, which also explains how the program works (MAYER 2007)	- performs exhaustive searches	- unnecessarily bulky output
TRED	(SOKOL *et al.* 2007)	- performs exhaustive searches	- the input file should have no header

2.3.1 Pre-selection

Initially, I took into consideration all repeat finding programs which included the motif size range of microsatellites among their tandem repeat finding capabilities. However, due to the high abundance of microsatellite hits, and the more specialized parameters required to fine tune the tolerance of imperfection within smaller tandem repeats, the search is

usually not efficient enough if repeats of all sizes are searched for at the same time. **Figure 2.17** shows a categorization of repeat finding programs based on their main utility. All the programs outside the circles can find large as well as small repeats in all possible orientations. The circles contain the programs which find exclusively tandem repeats, separated based on their capacity to tolerate imperfections within the tandem repeat hits: perfect tandem repeat finders and approximate tandem repeat finders.

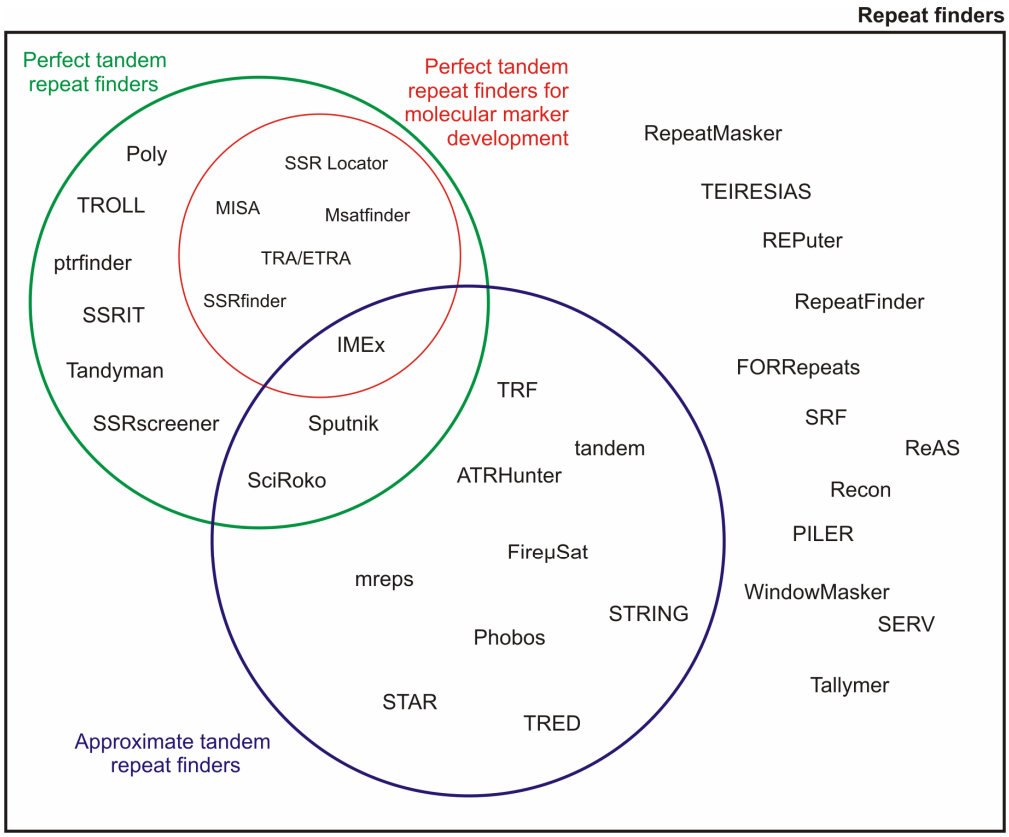

Figure 2.17: Categorization of repeat finders based on their main use**.** Within the coloured circles are those programs which specifically find tandem repeats and are therefore potentially useful for microsatellite searches. Red= to search for perfect tandem repeats in small sequences (<100000 nt), green= to search for perfect tandem repeats in large sequences, blue= to search for approximate tandem repeats. Outside the circles are programs useful for finding larger repeats, both tandem and interspersed ones. The citations for the tandem repeat finders can be found in **table 2.3**, and the ones for the remaining repeat finders are in **table 2.1**.

2.3.1.1 Perfect tandem repeat finders

Most of the earlier programs for microsatellite search were written with the specific purpose of scanning database entries for microsatellites or minisatellites to develop molecular markers. Therefore, it was not necessary to perform exhaustive searches, and the query sequences were also expected to be short (<100000 nt). These programs are grouped in a red circle in **figure 2.17**; they look only for perfect microsatellites, or allow a very limited amount of imperfections, but have no statistically based method to assess the significance of imperfect microsatellite hits. Additionally, these special-purpose programs are coupled with 'helper tools' or accessory scripts to retrieve flanking sequences to design PCR primers for the markers.

The remaining perfect tandem repeat finders in the green group (**figure 2.17**), except for the Perl scripts ptrfinder and SSRIT, can process larger DNA sequences (hum22.fa of ~49 Mb). There were small differences among the results because of two factors: heuristics and the inclusion of partial motifs as part of the hits. The programs TROLL and Poly use heuristics in their algorithms, which means that they randomly miss some hits during the search. The amount of hits missed is proportional to the length of the sequence. A summary of the main features of the programs in this category is presented in **table 2.6**, and additional remarks about some of the programs (the ones which were expected to be potentially useful for parts of my project) are presented below.

Table 2.6: Main features of programs which search only for perfect tandem repeats

Program	Publication	Outstanding Characteristics	Drawbacks
Tandyman	*NP (LEACH and CLELAND 1997)	- Exhaustive - It can look for repeats in both, DNA and aminoacid sequences	- Can not process large sequences (hum.22.fa) - Reports redundant hits in the output
SSR screener	*NP (GUR-ARIE *et al.* 2000)	- Good speed (0.14 Mbases/sec)	- The input modality is slow and error-prone - No parameter to specify the maximum motif length - The output is unnecessarily bulky
SSRIT	*NP (TEMNYKH *et al.* 2001)	- Exhaustive - Allows to specify minimum length threshold independently for each motif	- Can not process large sequences (hum.22.fa) - Looks only for motifs 2 to 4 nt long - Does not report partial repeat units

TROLL	(CASTELO *et al.* 2002)	- Relatively fast	- Not exhaustive - Can not handle soft-masked sequences - Counts characters in fasta header as part of the sequence - Does not report partial repeat units
Poly	(BIZZARO and MARX 2003)	- It offers a quantitative analysis of microsatellites	- Not exhaustive - Slow - Does not report the positions of the microsatellites, it only counts the occurrences of tandem repeats for each motif independently
ptrfinder	*NP Author: Jack R. Collins (COLLINS *et al.* 2003)	- Exhaustive - Suitable to run on a parallel system	- It failed to process large sequences (NC_003997.fa, hum.22.fa) although it should be capable of this.

*NP : No scientific journal publication was available describing the algorithm. Therefore I cite directly the authors and/or the application paper where the program was first used.

TROLL

The program TROLL has a number of bugs (as described below) but, once these are overcome, it is efficient and can run on complete eukaryotic chromosomes. A search for microsatellites with motifs from 1 to 5 nt in length on the human chromosome 22 (hum22.fa) took approx. 12 minute. Some important faults to watch out for when using this program are: the sequence in the query file should be in the same case as the motifs in the 'motifs file', the program will ignore characters in different case, which means that it can not be used to scan soft-masked sequences. TROLL also does not recognize FASTA headers, counting them as part of the sequence. Finally, the motif file should be saved in unix format and should contain one motif per line, without any spaces. Otherwise, the search can get aborted or produce incomplete results (lacking some motifs).

Poly

This program is not a repeat finder as is sometimes assumed (SHARMA *et al.* 2007). It reports the representation of tandem repeats available in a sequence for a specified motif length. It generates one file per motif listing the number of occurrences of this motif with 1, 2, 3,… up to n repetitions in the sequence. In two additional columns it gives the logarithmic values required to construct the frequency plots per motif, as shown in the author's publication (BIZZARO and MARX 2003). Since the program does not output the positions of the repeats found, it is not useful to analyze microsatellite distribution.

ptrfinder

This program presented several faults during the initial test runs. It did run fine on the smaller test sequences zubeca.fa (1400 nt) and danio.fa (13442 nt), but it crashed showing a 'segmentation fault' error when running NC_003997.fa (a smaller 17400 nt segment of the same sequence was processed without problems). This is probably a bug because the program is supposed to be able to run on large sequences (COLLINS *et al.* 2003). The search results show that the program reports repeats with complete as well as partial motifs (i.e. two and a half repeats). The reported motifs correspond to the lexicographic equivalent of the motif found, as shown in figure 2.18.

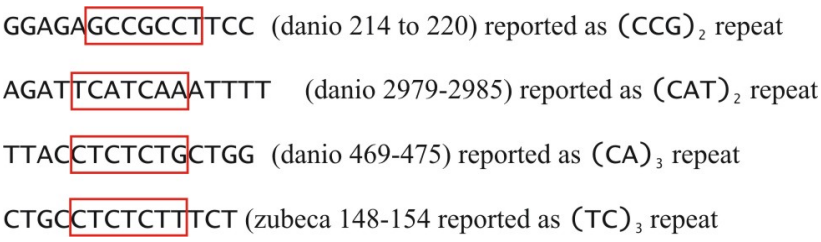

Figure 2.18: Example of hits obtained with ptrfinder. The motif reported by the program corresponds to the lexicographic equivalent of the tandem repeat in the sequence, which can not be directly recognized when observing the tandem repeat sequence according to its coordinates (shown in a red box). Therefore, ptrfinder results can not be directly compared with other program's results.

2.3.1.2 Approximate tandem repeat finders for genome-wide analysis

Approximate tandem repeats are based on complex algorithms for the detection of degraded copies of tandem repeat units. All programs shown in **table 2.7** (page 78) were developed for processing large genomic sequences, and all but STRING can run through large and small sequences without problems related to the length of the sequence *per se*. The program STRING appeared to be adequate for the search of microsatellites. However, it does not have an option to limit the maximum motif size, and it searches therefore for all possible motif lengths. In the case of searching only for microsatellites, the lack of such option renders the program highly inefficient, and the majority of the search hits would have to be filtered out after the search. There might be a way to modify this upper motif length in the C code, but this is not easy to infer from the program code or its paper (PARISI *et al.* 2003).

The program Sputnik does only run through large sequences when the option -A is included in the search parameters. This option (see **figure 2.4** for parameter details) sets the amount of recursion performed by the search algorithm automatically depending on the motif length, and it also adjusts the score for the first repeat unit automatically. During the tests, Sputnik reported satisfactorily the microsatellites in zubeca.fa with parameters -v 1 -u 5 -m 2 -n -6 -s 8 -p -L 16 -l -1 when not using the -A option, but it missed some hits with the same parameters when including it. This may be due to a reduction in recursion without which, however, it would not be possible for the Sputnik algorithm to complete searches through longer sequences. Nevertheless, this option also induces some aberrant behaviour in the program when the score and minimum length parameters are too low (see **figure 2.19**). Therefore, the score and minimum length parameters need to be adjusted accordingly when searching through large sequences.

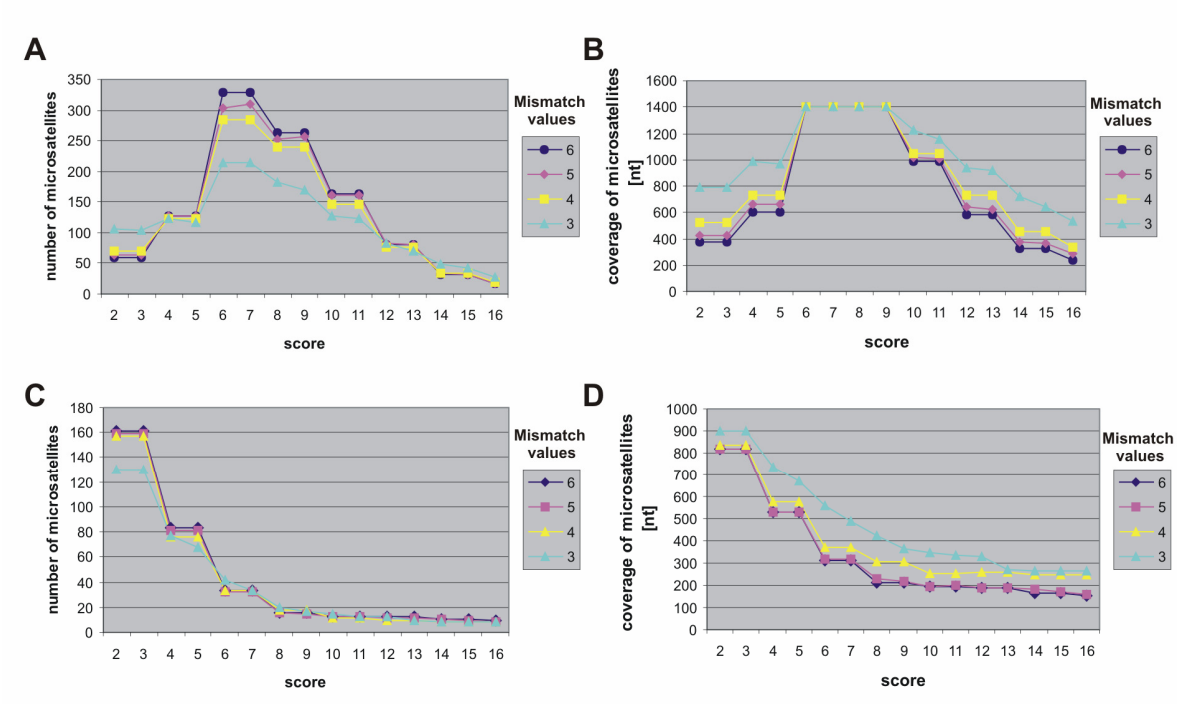

Figure 2.19: Screen of variation in Sputnik output on zubeca.fa by varying the parameters for mismatch penalty '-m' and score '-*s*' (parameters -v 1 -u 5 -m *y* -n -3 -*s* -*x* -p -L 4 -l). The mismatch penalty values are given to the program as negative values, but the absolute values are shown here. The results in graphs A and B, were obtained by adding the -A option to the parameters mentioned, while the bottom graphs were produced without it. Microsatellite content is represented in both number of microsatellites (left) and nucleotide coverage of these (right). By comparison of the top and bottom graphs, the use of the –A option produces aberrant results for the range of scores below 12. The upper plateau reached in graph B corresponds to the length of the query sequence, 1400 nt. and the empirical coverage of the microsatellites in the sequence is 266 nt. Similar behaviour is observed for higher minimum length values (-L) and other test sequences.

In terms of input parameter flexibility, all programs except STRING and STAR offer the user some control over the search. For the program STRING the desired adjustments may be done in the C code, but the variables are difficult to make sense of without C programming knowledge, and there is no mention of the value ranges to use, rendering the parameter optimization a trial and error process. The program STAR, on the other hand, is based on a compression algorithm, and it assesses automatically the significance of the repeats based on their compression rates.

A comparison of input parameters among the tandem repeat finders tested here can be seen in **table A1** in the Appendix section. The programs Sputnik, TRF, ATRHunter, and tandem, as well as the newer programs Phobos and TRED, offer a large range of parameters and therefore great flexibility for defining the microsatellite search. The programs mreps and IMEx have a rather reduced set of parameters: mreps bases the control of the degree of imperfection in microsatellites on its 'resolution' parameter, while IMEx offers a choice of maximum imperfection as absolute and as a percentage value. The programs TRF, ATRHunter, and tandem, have almost identical parameters because they are based on very similar algorithms. These algorithms divide the search in two phases: a detection phase where candidate tandem repeats are identified based on heuristic processes, and a verification phase based on a wraparound dynamic programming algorithm for comparing the hits with a perfect tandem repeat of the consensus motif determined for each of these. The differences among the programs reside in the detection phase, mainly in the way the scores and the match and indel probabilities are calculated. In the case of the program tandem, the heuristic process reduces the window comparison to immediately adjacent ones in order to reduce the processing time (DOMANIC and PREPARATA 2007). Both ATRHunter and tandem are supposed to be faster than TRF, but this comes with a reduction in output information (see **figure 2.20**). All three programs do also produce alignment files in html, but this option can only be suppressed in TRF. The option of suppressing large output files, as is the case of html alignment files, and the possibility to reduce the output to summary tables, are essential program features for large scale genome analyses. Otherwise the program invests resources in building files that will probably not be used due to the large scale of the analysis, and the results occupy an excess of storage space, which can be an important limiting factor.

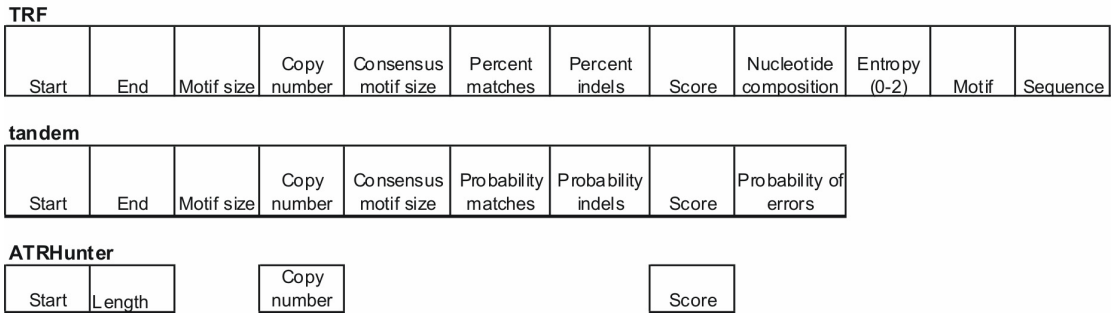

Figure 2.20: Comparison of output information among TRF, tandem, and ATRHunter**.**

The programs Sputnik and mreps showed problems when processing sequences containing gaps (N characters): hum22.fa and danio.fa. The program mreps replaces any N's in the query sequence with random nucleotides during the search, but gaps like the one in the danio.fa sequence with 341 N's are already too big for the program causing an abortion of the search ("Error: Too many N's in the window"). In the case of Sputnik, the search is carried out without problems in sequences containing N's, because the program is supposed to ignore N characters. If, however, a microsatellite occurs immediately before a run of N's, the program reports the gap as part of the microsatellite. Gaps in genomic sequences flanked by microsatellites are very common probably because the microsatellite sequence was the reason for obtaining low quality reads in the gap region, which were therefore replaced by N's. Therefore, in many cases it may be well justified to extend the microsatellite through the gap, especially if the same kind of microsatellite is found at the other end of the gap. However, microsatellites are not the only problematic features for DNA sequencing, and it is best to ignore gaps as a whole during microsatellite search.

Finally, the capacity of automation is a fundamental feature first, for the optimization of search parameters previous to the utilization of the program (see Chapter IV), and second, to have the option of performing serial or pseudo-parallel search runs in all sequences corresponding to a genome, or to a whole database. However, not many programs had this option, either because only a graphical interface was available, or because the command line versions would prompt for program parameters only after the program invocation. This and other issues were very common among tandem repeat finders, and therefore I present a description of "common problematic characteristics" in the next section. A summary of these problems is also presented in **table 2.7** for all approximate tandem repeats.

Table 2.7: Problematic characteristics observed in approximate tandem repeat finders

Program	Publication	Problematic characteristics
Sputnik II	*NP (LA ROTA *et al.* 2005)	- Extends repeats through gaps. - Treats substitutions and indels in the same way (same penalty for both). - Does not extend through imperfections efficiently.
TRF	(BENSON 1999)	- Not very fast. - Includes redundant hits in the output.
STRING	(PARISI *et al.* 2003)	- The upper limit for the motif size can not be specified.
mreps	(KOLPAKOV *et al.* 2003)	- Its algorithm is based on Hamming distance calculation. - Can not handle gaps in the DNA sequence. When a short amount of Ns are present along the sequence, the program replaces these for random sequences. A gap as big as the one in the danio.fa sequence (314 bp) produces an error which halts the program. - Includes redundant hits in the output. - Does not report motifs, it is specialized in detecting tandem periodicity within a sequence, not specific motifs.
STAR	(DELGRANGE and RIVALS 2004)	- Produces one output file per motif and redundancy can exist among files. - Does not allow parameter modifications. - Output file too bulky.
ATRHunter	(WEXLER *et al.* 2005)	- Prompts for input parameters one by one after the program invocation (9 prompts in total). - Output in txt and html formats. No possibility to suppress the html formatted output.
IMEx	(MUDUNURI and NAGARAJARAM 2007)	- Prompts for input parameters one by one after the program invocation (18 prompts in total). - Output in html and distributed in two files per input sequence: positions file and alignments file. The alignment file can not be suppressed.
SciRoKo	(KOFLER *et al.* 2007b)	- Only available in GUI version ‡. - Treats substitutions and indels in the same way (same penalty for both)
tandem	(DOMANIC and PREPARATA 2007)	- Only available in GUI version. - Output files are unnecessarily bulky. - Does not count the N's in the query sequence and therefore the output positions are wrong if the query contains gaps. - No motif or microsatellite sequence in output. - Output in txt and html formats. No possibility to suppress the html formatted output. - No possibility to rename the output files.-

*NP : No publication was available describing the algorithm. Therefore I mention the authors and/or the application paper where the program was first used.
‡ : The author Robert Koffler made available a command line version (SciRoKoCo) on request.

2.3.1.3 Common problematic characteristics

Here I describe several problematic characteristics repeatedly observed in microsatellite finding programs. All characteristics mentioned below, except the output of redundant hits,

seem to be generic problems in programming practice. Moreover, many of these problems come about as a consequence of the programmer's effort to make the program easy to use or self-explaining. When possible, these program issues should be avoided or corrected in future versions of the programs, or in new programs.

Non-automatable input modality

For a tandem repeat finder to be useful for studying whole genomes some degree of automation is usually necessary. Genome sequences are available as drafts from which new versions are published, often on a yearly basis (KAROLCHIK *et al.* 2003). Therefore, the same analysis will need to be carried out every time a new draft version is released, so that the annotations can be updated. Nonetheless, several of the programs with the capacity of analyzing large DNA sequences have not-automatable input modalities. This was either because the program was only available as GUI version (tandem, SciRoKo) or because the command line version required several input steps (ATRHunter, IMEx) as seen in **figure 2.21**. On March 21, 2009, a new version of IMEx was released, which is available in their web page (http://210.212.215.200/IMEX/news.html). This new version should allow the submission of batch files and the search of compound microsatellites. IMEx 2.0 was not tested for the present project.

Another feature hindering automation is the inability to re-name the output files (e.g. the programs tandem and TRA/ETRA). It is usually necessary to be able to name the output with the search settings when optimizing the program's search parameters. The program TRF, for example, does not allow to give specific output names either, but it automatically includes the parameter settings used for the search in the output file's name, which is very convenient for further analyses.

A

```
C:\Documents and Settings\imj15\Desktop\prog\ATRhunter>atrhunter_windows.exe
NC_003997.fa
Please insert the desired alignment parameters (whole positive numbers)
Match score: 2
Mismatch penalty: 3
Gap (insertion or deletion) penalty: 6
Terminal gap penalty: 5
Please insert the desired maximum motif length (1-500): 6
Please choose the desired definition of an ATR
1 - Using the similarity level between adjacent copies
2 - Using the average similarity level between adjacent copies
3 - Using the alignment score with a repeating pattern
Definition (1-3): 3
Minimum Similarity Level: (minimum - 50; maximum - Match Score*100):75
Do you want ATRhunter to generate a file with the alignments (Y/N)?n

Do you want ATRhunter to generate the output also as a text file (Y/N)?y
```

B

```
ENTER THE 'k' VALUES (Imperfection limit/repeat unit):
Mono [0-1]:
Di [0-2]:
Tri [0-3]:
Tetra [0-4]:
Penta [0-5]:
Hexa [0-6]:

ENTER THE 'p' VALUES (Imperfection percentage):
Mono [0-90]:
Di [0-90]:
Tri [0-90]:
Tetra [0-90]:
Penta [0-90]:
Hexa [0-90]:

ENTER THE 'n' VALUES (number of repeat units/tract):
Mono:
Di:
Tri:
Tetra:
Penta:
Hexa:
```

Figure 2.21: Example of the query submission for program ATRhunter (A) and IMEx (B). This kind of input style where parameters need to be specified one by one to the program poses an obstacle to automation of search tasks. Moreover, if the program needs to interact 10 or more times with the user per query submitted, the error rate during query submission becomes exponential, and frustrating.

Bulky output format

The programs ptrfinder (COLLINS *et al.* 2003) and SSRscreener (GUR-ARIE *et al.* 2000) produce unnecessarily large output files by including redundant explanatory text in each line (**figure 2.22**). The program STAR outputs a whole block of text per microsatellite identified (**figure 2.23**), and the program tandem (DOMANIC and PREPARATA 2007) does even include comparisons for every pair of adjacent repeat units conforming a reported tandem repeat (**figure 2.24**). The program reports 19238 microsatellites for the hum22.fa sequence and the output contains 2398379 lines, and an additional html file with the same

information is also produced in the output and can not be suppressed. Html files as only output format, as in the case of the program TRA/ETRA (BILGEN *et al.* 2004), or as auxiliary files which can not be suppressed, as is the case of IMEx (MUDUNURI and NAGARAJARAM 2007), ATRHunter, and tandem, can also be considered as excessively bulky output.

ptrfinder:

```
>ZuBeCa fraccion   Repeat: PATTERN tc LENGTH 2 TIMES 2 START 10 STOP 14 ID 2
>ZuBeCa fraccion   Repeat: PATTERN ag LENGTH 2 TIMES 2 START 77 STOP 81 ID 3
>ZuBeCa fraccion   Repeat: PATTERN ct LENGTH 2 TIMES 2 START 125 STOP 129 ID 4
>ZuBeCa fraccion   Repeat: PATTERN tc LENGTH 2 TIMES 2 START 130 STOP 134 ID 5
```

SSRscreener:

```
1-nucleotide SSR motif: A appears 3 times at position 6.000000 bp
1-nucleotide SSR motif: A appears 4 times at position 11.000000 bp
1-nucleotide SSR motif: G appears 3 times at position 25.000000 bp
1-nucleotide SSR motif: T appears 3 times at position 48.000000 bp
```

Figure 2.22: Examples of bulky output with explanatory text in each line from the programs ptrfinder (COLLINS *et al.* 2003) SSRscreener (GUR-ARIE *et al.* 2000)

```
ZONE      1 BEGIN_POS      2206 END_POS      2276 LG      71 GAIN     19
A     36 C      2 G      0 T     33 N      0 %AT  97 %GC   2 Biais GC 1.00
Phase 1
Consensus columns, counts of matches, substitutions, and deletions
Position     1    2    3
Nb_Match    18   18   18
Nb_Subst     0    0    0
Nb_Del       0    0    0
Insertion columns number: 1 and list of positions and counts
Position     1
Nb_Ins      17
         Match  Sub  Del  Ins
Totals      54    0    0   17
Percents  76.1  0.0  0.0 23.9
Nb_Motifs 18.00 Percent_of_exact_motifs  5.56 Consensus 0 atat
```

Figure 2.23: Bulky output of STAR. A block of data like the one depicted here is given in the output file for each repeat detected by the program STAR (DELGRANGE and RIVALS 2004).

```
Repeat start = 55, end = 70, period = 4

Length of Consensus Pattern = 4
GATC

Copy Number = 4

Alignments of Each Copy with Consensus Pattern

Length of 1. copy = 4
GATC
GATC
Matches = 4 Mismatches = 0 Insertions = 0 Deletions = 0 Score = 8

Length of 2. copy = 4
GATC
GATC
Matches = 4 Mismatches = 0 Insertions = 0 Deletions = 0 Score = 8

Length of 3. copy = 4
GATC
GATC
Matches = 4 Mismatches = 0 Insertions = 0 Deletions = 0 Score = 8

Length of 4. copy = 4
GATC
GATC
Matches = 4 Mismatches = 0 Insertions = 0 Deletions = 0 Score = 8

Total Matches = 16 (100.00%) Total Mismatches = 0 (0.00%) Total Indels = 0
(0.00%)

Alignments of Adjacent Copies

1. copy vs 2. copy
GATC
GATC
Matches = 4 Mismatches = 0 Insertions = 0 Deletions = 0 Score = 8

2. copy vs 3. copy
GATC
GATC
Matches = 4 Mismatches = 0 Insertions = 0 Deletions = 0 Score = 8

3. copy vs 4. copy
GATC
GATC
Matches = 4 Mismatches = 0 Insertions = 0 Deletions = 0 Score = 8

Total Matches = 12 (100.00%) Total Mismatches = 0 (0.00%) Total Indels = 0
(0.00%)
```

Figure 2.24: Bulky output of tandem. For each tandem repeat reported by the program tandem (DOMANIC and PREPARATA 2007), individual repeat unit comparisons like the one above are included below the summary table in the same text file. This particular block describes the repeat unit comparisons for a tetranucleotide with four repetitions. The program ATRHunter reports similar repeat unit comparisons in its html output.

As can be observed, bulky output files are a very common characteristic among tandem repeat finders. This should not be a reason to discard a good program. However, it can be a

problem when the storage space is limited, and oftentimes it supposes the necessity of additional processing before the information can be analyzed.

Redundant hits in the output

The programs TRF , ATRHunter, tandem, Tandyman, and STAR report redundant hits in the output. In the case of STAR one output file is produced per microsatellite motif, therefore the redundant hits are not in the same file. Reporting redundant hits is a common side effect of tandem repeat finders. Usually in order to find tandem repeats with different motif sizes, the query sequence gets scanned independently for each motif size. During this process the same repeat can be reported several times with different motif sizes (for example a dinucleotide AT could also be detected as a tetranucleotide ATAT if it is long enough). Programs then filter the redundant hits out by keeping only the hit with the smallest motif and/or the best score. From a biological point of view, it may make sense to keep at least one of the redundant hits, because slippage mutations may occur based on either of the motifs. It could be argued that a shorter motif (e.g. AT) is more likely to mutate than a longer one (e.g ATTA) because, in a sequence of equal length, the shorter motif has more repeats than the longer one, and therefore more possibilities to undergo strand slippage during replication (BROHEDE *et al.* 2002; ELLEGREN 2000). However, longer motifs have a stronger influence on cellular repair systems because they can form larger loops (JENSEN *et al.* 2005). Consequently, it can not be generalized that the shortest motifs will be the most mutable, and therefore the best choice to represent a microsatellite. In Chapter III I present a java program for filtering redundancy from microsatellite datasets by joining redundant hits and keeping the originally reported motifs for further consideration.

2.3.2 Program benchmarking and comparison

Before proceeding with the comparison of search results among different programs repeat finding programs, it is important to define a meaningful measure to express these search results. Comparisons among the output of different microsatellite finding programs are always expressed and compared in terms of number of hits in the output (for example see (KOLPAKOV *et al.* 2003; LECLERCQ *et al.* 2007; MUDUNURI and NAGARAJARAM 2007; SHARMA *et al.* 2007; WEXLER *et al.* 2005)). However, for several reasons that I will discuss here, the comparison of absolute microsatellite numbers generated by different programs can lead to biased observations and conclusions.

Initially, in order to characterize the search capacity of the programs, I constructed microsatellite number and coverage distribution graphs for each program. These consisted on databases of search results obtained in serial program runs with all possible search parameter value combinations, varying one parameter at a time. A thorough testing of parameters was, however, only possible for those programs with automatable executables: TROLL, Tandyman, Poly, Sputnik, TRF and SciRoKoCo (command line version of SciRoKo). I subsequently compared these output distributions to observe program performance and to find overlapping regions within the distributions among programs.

2.3.2.1 Perfect microsatellite searches

When only perfect microsatellites are analyzed, counting and comparing 'numbers of hits' can be a valid option, but only if the same definition of microsatellite is used by each program involved. The microsatellite hits sought after are defined to the program by setting parameter values like minimum and maximum motif length, the minimum microsatellite length, and the minimum score to be ahieved. **Figure 2.25** shows an example of microsatellite number distributions obtained for various perfect tandem repeat finders with a range of minimum repeat length values. It can be observed that the number of microsatellites in DNA sequences decreases exponentially as the minimum microsatellite length is increased. When not limited by the score, the minimum length of the microsatellites is the most influential parameter in microsatellite searches, and slight changes in the lower range of this length threshold can therefore produce large differences among the results (this was also observed by LECLERCQ *et al.* 2007). Furthermore, different ways of measuring this minimum length can produce artifactual differences among programs.

The minimum microsatellite length can be expressed in four different ways: as a single value for all microsatellite motif lengths measured in number of nucleotides, as a single value measured in numbers of repeats, or as individual values for each motif length, both in number of nucleotides and in number of repeats. As shown in the comparison among **figure 2.25 A** and **figure 2.25 B**, searches defined based on different measures of minimum numbers of repeats can produce different microsatellite counts even when only perfect microsatellites are searched for. However, if these definitions are equalized properly, the number of microsatellites found by different programs in the same query sequence should be identical or at least very similar. Any difference found among program outputs

after assuring the microsatellite definitions used are equivalent, are real program-related differences.

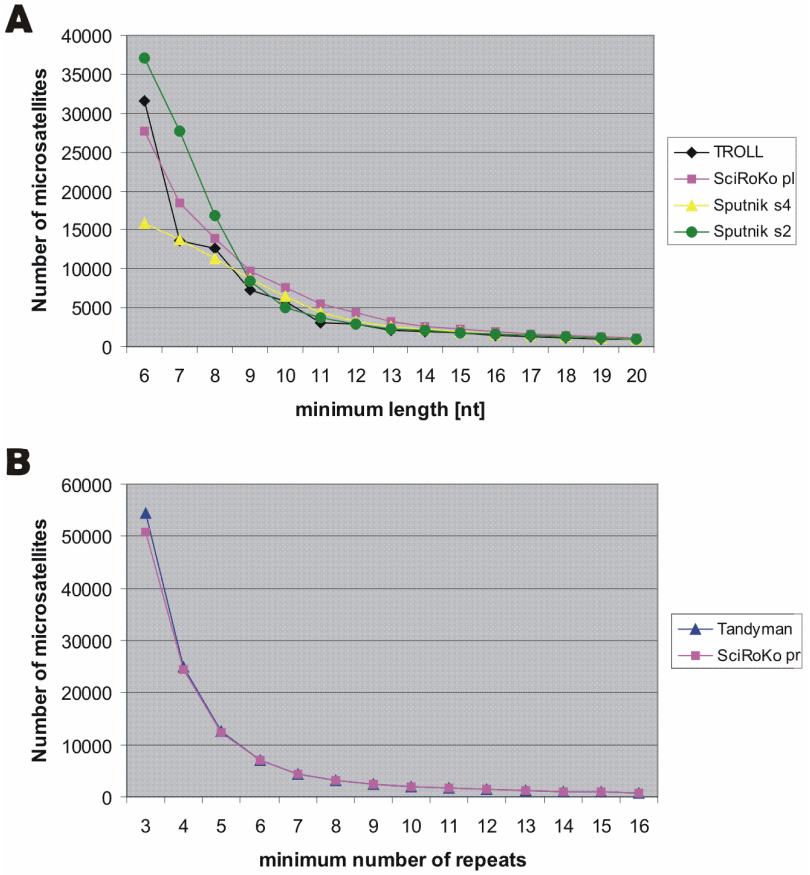

Figure 2.25: Comparison of the perfect microsatellite number distributions obtained with four different programs on the chromosome 1 of *Plasmodium falciparum* (test sequence plas1). Graph A shows the programs TROLL, SciRoKo in pl mode, and Sputnik with two different score values (2 and 4). These programs or program modes define the minimum microsatellite length in number of nucleotides, while the programs or program modes in graph B define the minimum microsatellite length in number of microsatellites. The first one is an absolute definition and the second one is relative to the motif length, therefore the results are different. The second definition is also more accurate, and therefore the microsatellite number distributions are almost identical. The minimum microsatellite length can also be assigned independently for each motif (with programs like MISA, SciRoKo in MISA mode, SSRIT, and IMEx), in which case the graph would have six dimensions, and the results would also not be directly comparable to the distributions in graphs A and B.

Additional sources of variation in the search results of perfect tandem repeats are: the use of heuristic or exhaustive algorithms, the method used for filtering redundancy, and the capacity of some programs to report partial motifs as part of microsatellite hits. Heuristic algorithms like TROLL or Poly usually miss some hits, and the number of missing hits is positively correlated with the length of the query sequence. Most perfect repeat finders

except Tandyman filter out redundancy. This redundancy can artefactually increase the number of hits in the output. Redundancy is, however, more of a problem and more difficult to filter out in imperfect repeat search results.

The reporting of partial motifs should not be a problem when only numbers of perfect microsatellitesare anlyzed. However, differences will arise when comparing microsatellite lengths and/or positions among the output of programs which report partial repeats (e.g. Tandyman, ptrfinder, SciRoKo with pl mode) and those that only report complete repeat units (e.g. TROLL, MISA, SSRIT, SSRscreener, Poly). The accumulation of small individual differences among microsatellite hits can produce considerable differences in the estimated total length or coverage of microsatellites in the query sequence. This is also more of a problem when analyzing imperfect tandem repeat search results because, in contrast to perfect microsatellites, start and end positions for imperfect microsatellites cannot usually be unambiguously assigned.

Leclercq *et al.* (2007) presented a comparison of algorithms for perfect tandem repeat detection among the programs mreps, TRF, Sputnik, STAR, and RepeatMasker. They found an 80-fold difference between the two extreme values returned by Sputnik and RepeatMasker. However, their comparisons were flawed in many ways. First, they used five programs based on very different algorithms and their comparisons were based on only one set of parameters per program. Additionally, TRF, mreps, and STAR are not suited for finding only perfect repeats; their results will always contain some imperfections. Sputnik has the option to report only perfect microsatellite hits, but it was apparently used with a mismatch penalty value (-6), which is only necessary for imperfect microsatellite search runs. Finally, RepeatMasker is not a program to search for microsatellites; it searches mainly for interspersed repeats and other repeat families contained in a consensus repeat library put together beforehand for each species (JURKA *et al.* 1992; SMIT *et al.* 1996-2007). Therefore, Leclercq *et al.* (2007) were comparing apples with pears and, although their discussion on parameter characteristics and program differences is in general correct, the numerical estimates they give are meaningless.

Another independent microsatellite search program comparison was carried out by Merkel and Gemmell (2008) on the programs TROLL, TRF, Sputnik, Msatfinder and mreps. Here again, perfect and approximate tandem repeats are mixed in the comparisons and the program's parameter settings used for the comparisons show that different microsatellite definitions were used in all cases.

Other than the two mentioned studies, the papers for the programs TROLL (CASTELO *et al.* 2002), ATRHunter (WEXLER *et al.* 2005), IMEx (MUDUNURI and NAGARAJARAM 2007), and tandem (DOMANIC and PREPARATA 2007) present limited program comparisons for the introduction of each new program. Castelo *et al.* (2002) focus only on execution time when comparing TROLL, TRF and Sputnik, and do not mention the program's search paramters. Mudunuri *et al.* (2007) also focus on execution times and compare numbers of microsatellites obtained with IMEx and TRF. Wexler (2005) compares ATRHunter with TRF and TEIRESIAS (one of the pioneer general purpose tandem repeat finders developed at IBM, RIGOUTSOS and FLORATOS 1998), and do also not specify the parameter sets used. Finally Domanic and Preparata (2007) compare 'tandem' with TRF and ATRHunter, they mention the parameters used (except for tandem), and they try to equalize the search parameters. But the comparisons are still done based on numbers of repeats and execution time, and only one parameter setting per program is used.

2.3.2.2 Approximate or imperfect microsatellite searches

In addition to the factors which can produce variation among search results in perfect microsatellite finders, the results of approximate microsatellite finders are strongly dependent on the program's algorithm and on the way microsatellite imperfections are defined. Furthermore differences in microsatellite structure complexity and imperfection can affect the absolute number of microsatellites reported by a program. Reporting only the number of microsatellites found implies losing the information on the length of microsatellites. A more informative measure to summarize microsatellite abundance information, especially for comparative genomics studies, would be the nucleotide coverage of microsatellites, because it depends more on the level of imperfection and motif structure of microsatellites than on the program used to identify them. The raw numbers and nucleotide coverage of microsatellites show only weak a correlation (**figure 2.26**) and, in the case of approximate tandem repeats, this can turn into an inverse correlation because adjacent perfect tandem repeats separated by small gaps can be joined into longer microsatellites when the search parameters are relaxed, reducing the number of hits reported but increasing the nucleotide coverage of these. Therefore it is important to report both measures, microsatellite hit number and nucleotide coverage, from analyses of microsatellite abundance.

Throughout this document I express microsatellite abundance results as both, number of microsatellites per Megabase (Mb) and nucleotide coverage in percentage, to account for

sequence length differences. I will refer to both measures together as the density of microsatellites.

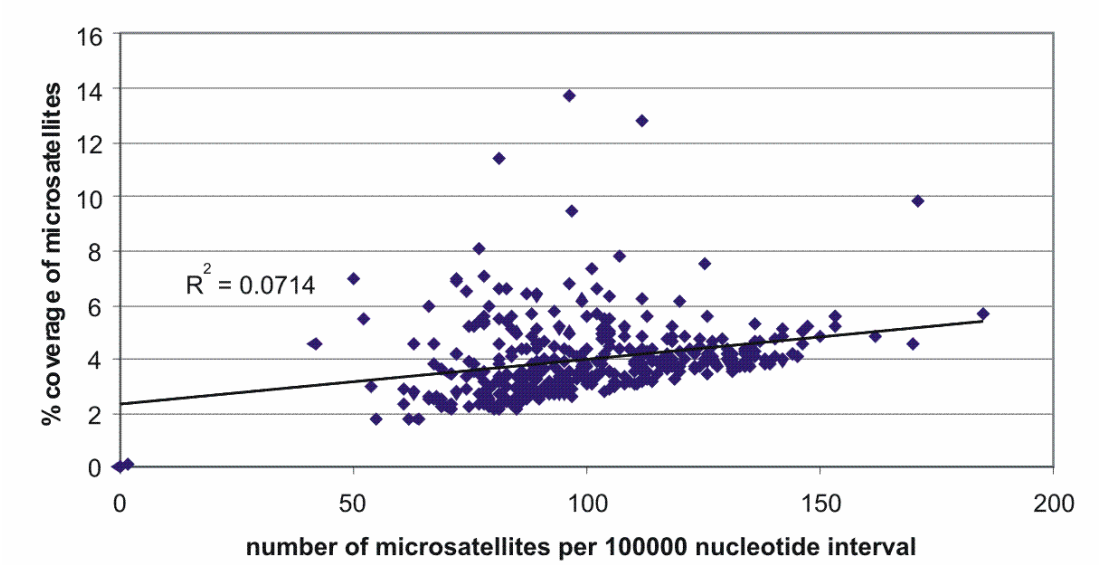

Figure 2.26: Correlation curve among two measures of microsatellite abundance; the number of microsatellites *vs* percent coverage of microsatellites. The data in this graph corresponds to human chromosome 22, and was obtained by running the program TRF with parameters 2 3 5 80 10 30 6 (these are relatively conservative values so to avoid false negatives). Each dot represents a non-overlapping sequence interval of 100000 nucleotides where microsatellites were quantified. The number of microsatellites and the coverage of these are positively correlated. However, only 7% of the variance is explained by the regression line.

Output distributions for each program were obtained by pooling the search results from serial runs with varying parameter settings for each program. The parameters covered gradually the range from very restrictive to very relaxed settings. By comparing these output distributions it becomes evident that there are usually regions within the distributions for each program that overlap with the distributions of other programs; after all, it is the same sequence that is being analyzed. These probably correspond to the parameter settings with which the program's microsatellite definitions converge. **Figure 2.27** shows boxplots comparing the output distributions for TRF, SciRoKo and Sputnik, where the distribution overlaps can be visualized. However, the major parts from the interquantiles do not overlap, showing that there is also considerable difference among program output ranges. When parameter settings corresponding to the non-overlapping regions of distributions are used for program comparisons, it is very likely that large differences will be found among programs. This would explain why Wexler *et al.* (2005) mentions that ATRHunter finds 61% (from comparisons of microsatellite numbers) more approximate repeats than TRF.

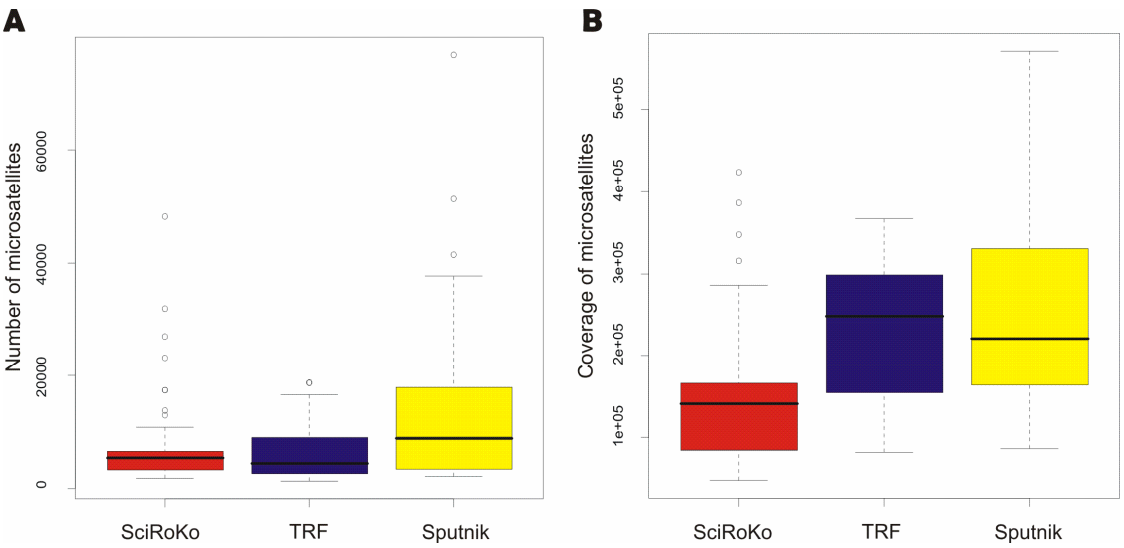

Figure 2.27: Comparison of microsatellite number and coverage distributions for SciRoKo, TRF, and Sputnik, on chromosome 1 of *Plasmodium falciparum* (test sequence plas1). The programs SciRoKo and TRF report fewer microsatellites than Sputnik, but the comparison of corresponding boxplots between graphs A and B shows that the hits reported by TRF are longer than from the other two programs. The real microsatellite coverage of only perfect microsatellite regions in this chromosome is 224571 nt. Imperfections included within the microsatellite hits will increase this coverage. But most of the coverage above 250000 nt in graph B corresponds to false negatives (too small or too interrupted microsatellies.

Based on Kruskal Wallis rank sum tests of the microsatellite coverage distributions, there are significant differences between SciRoKo and Sputnik (p-value = 7.305e-07) and between SciRoKo and TRF (p-value = 1.716e-06), while no significant differences exist among the coverage distributions of TRF and Sputnik (p-value = 0.6332). **Figure 2.28** shows comparisons of the programs with four representative parameter sets corresponding to the output distribution overlaps, so that the results have similar ranges. The program SciRoKo has by far the shortest interquartile space although its output numbers and coverage can reach the levels observed in the other two programs. However, with an increase in stringency in the parameters, these results drop rapidly in magnitude. This can be better observed in **figure 2.28.** The TRF output distribution, has an intermediate interquartile space and almost no outliers in **figure 2.27 A**, while the other two programs have multiple outliers. This may be because TRF bases the assessment of the imperfection content of microsatellite hits on match and indel probabilities which are expressed as proportions of the total microsatellite sequence length. This proportionality should minimize the amount of false negatives in the output. In the cases of SciRoKo or Sputnik, the allowed imperfection and other parameters to validate the microsatellite candidates are expressed in absolute

terms (except in SciRoKo's mmvp mode) and relatively extreme values can be used which produce a large number of false negatives (too short or too imperfect microsatellites).

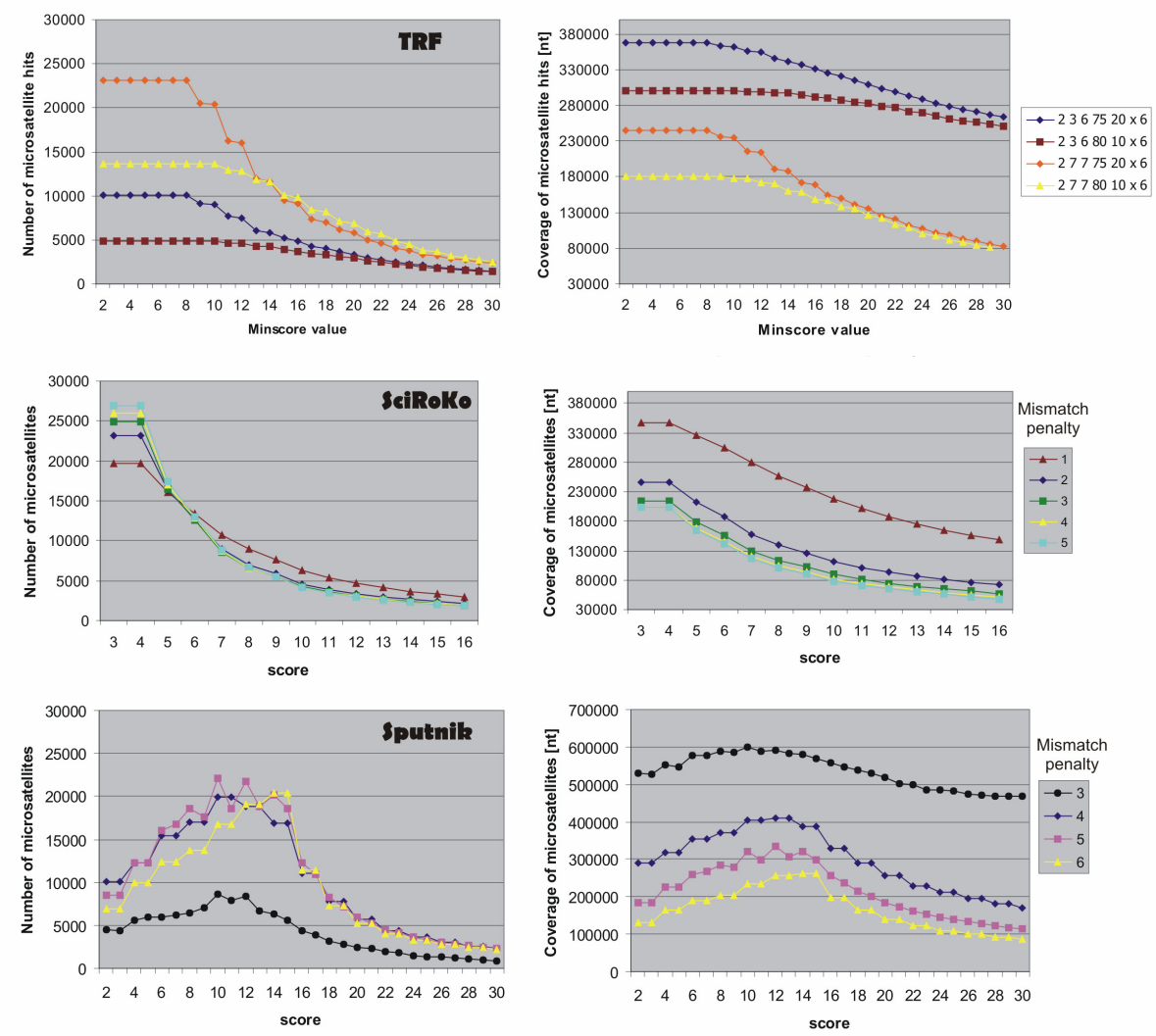

Figure 2.28: Comparison of number of microsatellites and microsatellite coverage distributions for TRF, SciRoKo, and Sputnik, each with 4 different parameter settings which are shown in the series names. The parameter sets are ordered from top to bottom with increasing stringency (the additional parameters for SciRoKo are: mmvp mode and a seedlength of 4, and for Sputnik: m 2 and L 8) , on chromosome 1 of *Plasmodium falciparum* (test sequence plas1). For each program, the results are expressed in number of microsatellites (left side graphs) and nucleotide coverage of these microsatellites (right side graphs). The parameter ranges used in these graphs allow direct comparisons of the programs because the definitions of microsatellites are similar. Note that the scale of the bottom right graph is smaller so it can fit the fourth curve.

The program Sputnik showed the highest dispersion in its boxplot even though the program only searches for motifs from 1 to 5 nt. However, these distributions are inflated by the aberrant behaviour it displays due to the automatic setting of recursion by the option –

A (mentioned in page 42). This can also be observed in the corresponding graphs in **figure 2.28**.

From these comparisons it is clear that not only the program, but the specific parameters used, can strongly affect the resulting microsatellite datasets. Therefore, the search parameters should always be specified when publishing analyses based on microsatellite searches, so that the results are susceptible to comparison and reproduction, if necessary.

2.3.3 Selected programs

Based on the benchmarking and comparisons, the programs which fit most of the requirements to be used in comparative genomics studies are TRF and SciRoKo (**table 2.8**).

Table 2.8: Programs with good potential for whole-genome microsatellite scans

Program	Publication	Outstanding characteristics	Drawbacks
TRF	(BENSON 1999)	- Extends efficiently through imperfections - Very flexible for parameter modification	- Not very fast. - Reports up to 5 redundant hits per microsatellite.
SciRoKo	(KOFLER *et al.* 2007b)	- Extremely fast - Very flexible for parameter modification - Offers statistical analysis of microsatellite datasets - Offers three different options of output format, a SciRoKo-specific one and the other two similar to sputnik, which makes results easy to process	- Treats substitutions and indels in the same way (same penalty for both)

SciRoKo showed the best speed performance for searching within whole eukaryotic chromosomes. Using default parameters the program can search through small chromosomes (i.e. 36 Mb) in two seconds and through the human chromosome X (~127 Mb) in 85 seconds. Initially, however, testing SciRoKo was a very tedious task because it was only available as a graphical interface. Therefore, files needed to be loaded manually and one by one, and then each output file had to be reloaded to be exported to the required format manually. After contacting the author Robert Kofler and explaining the problem, he made available a command line version SciRoKoCo, which includes all search-related parameters from the graphical interface and is therefore very easy to automate.

The microsatellite number and coverage distributions presented here are useful for preliminary comparisons of programs, and to select the appropriate parameters and parameter values for the optimization of microsatellite searches. Superposed microsatellite number and coverage distributions for TRF and SciRoKo are shown in **figure 2.29**. Once these are selected, a thorough comparison needs to be carried out by checking the output files for false positives, missing hits, and possible biases produced by the program's search algorithm, both in the detection of perfect as well as imperfect tandem repeats. The comparison and parameter optimization among the programs TRF and SciRoKo are presented in the next two chapters.

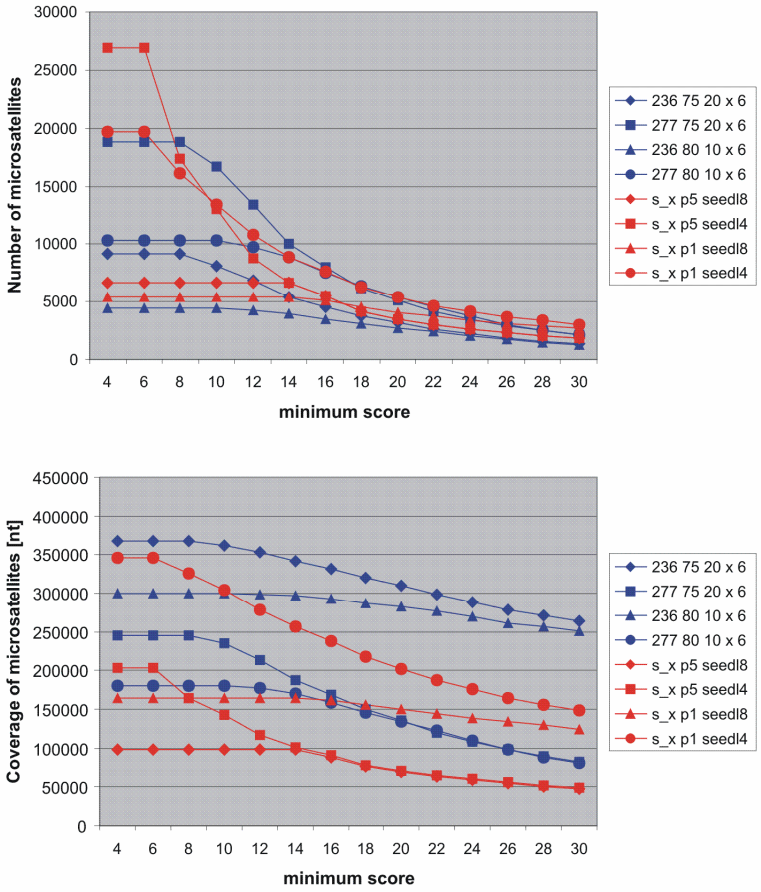

Figure 2.29: Comparison of microsatellite number and coverage distributions between TRF and SciRoKo. Four different parameter combinations are shown for each program, as specified in the series names. The parameter sets are ordered with increasing stringency from top to bottom (the additional parameters for SciRoKo are: mmvp mode and a seedlength of 4, and for SciRoKo: m 2 and L 8) , on chromosome 1 of *Plasmodium falciparum* (test sequence plas1).

2.4 Conclusions

For biologists, the main interest in finding microsatellites, usually with defined motifs or nucleotide composition, is to use them as molecular markers for one of their multiple applications (see Chapter I). Less widespread has been the interest to identify and study microsatellites for their own sake, so to understand the evolutionary processes they participate in. Accordingly, the majority of programs available to find microsatellites in DNA sequences are not suitable for unbiased whole-genome scans.

The few papers available on comparisons among tandem repeat finding programs presented biased results because no attempt was made to optimize the program's parameters to search for the same microsatellite characteristics (using equivalent microsatellite definitions). Moreover, comparisons were usually based on computation time, and on a algorithm-dependent measurement of output content: the total number of hits. Due to the complexity of microsatellite structure and the variation in program's algorithms, these comparisons are misleading.

Different tandem repeat finders can produce significantly different microsatellite datasets. However, these differences are strongly dependent on the search parameters, which can usually be adjusted through a range of values. It is therefore essential to make sure that equivalent search definitions are used when comparing programs. Parameters affecting directly or indirectly the minimum length thresholds of microsatellite hits have the highest influence on the number of perfect microsatellites reported.

No repeat finding program examined here shows all sought-after characteristics in one. However, the programs TRF and SciRoKo are the best heuristic options available to date. Both programs can process large sequences without apparent problems, while providing relatively complete datasets, easy to parse and informative output, and fully automatable user interfaces. The main directly observable differences among TRF and SciRoKo are the speed (SciRoKo is two orders of magnitude faster than TRF) and the capacity to extend through imperfections (TRF extends more efficiently through imperfections based on a probabilistic model,).

2.5 References

AHO, A. V., and M. J. CORASICK, 1975 Efficient string matching: an aid to bibliographic search. Commun. ACM **18:** 333-340.

AISHWARYA, V., A. GROVER and P. C. SHARMA, 2007 EuMicroSatdb: a database for microsatellites in the sequenced genomes of eukaryotes. BMC Genomics **8:** 225.

AISHWARYA, V., and P. C. SHARMA, 2007 UgMicroSatdb: database for mining microsatellites from unigenes. Nucleic Acids Res.

ANWAR, T., and A. U. KHAN, 2006 SSRscanner: a program for reporting distribution and exact location of simple sequence repeats. Bioinformation **1:** 89-91.

ARCHAK, S., E. MEDURI, P. S. KUMAR and J. NAGARAJU, 2007 InSatDb: a microsatellite database of fully sequenced insect genomes. Nucleic Acids Res **35:** D36-39.

BAO, Z., and S. R. EDDY, 2002 Automated *de novo* identification of repeat sequence families in sequenced genomes. Genome Res **12:** 1269-1276.

BENSON, G., 1999 Tandem repeats finder: a program to analyze DNA sequences. Nucleic Acids Res **27:** 573-580.

BENSON, G., and M. S. WATERMAN, 1994 A method for fast database search for all k-nucleotide repeats. Nucleic Acids Res **22:** 4828-4836.

BILGEN, M., M. KARACA, A. N. ONUS and A. G. INCE, 2004 A software program combining sequence motif searches with keywords for finding repeats containing DNA sequences. Bioinformatics **20:** 3379-3386.

BIZZARO, J. W., and K. A. MARX, 2003 Poly: a quantitative analysis tool for simple sequence repeat (SSR) tracts in DNA. BMC Bioinformatics **4:** 22.

BROHEDE, J., C. R. PRIMMER, A. MOLLER and H. ELLEGREN, 2002 Heterogeneity in the rate and pattern of germline mutation at individual microsatellite loci. Nucleic Acids Res **30:** 1997-2003.

CAMPAGNA, D., C. ROMUALDI, N. VITULO, M. DEL FAVERO, M. LEXA *et al.*, 2005 RAP: a new computer program for *de novo* identification of repeated sequences in whole genomes. Bioinformatics **21:** 582-588.

CASTELO, A. T., W. MARTINS and G. R. GAO, 2002 TROLL--tandem repeat occurrence locator. Bioinformatics **18:** 634-636.

CERESINI, P. C., C. L. S. P. SILVA, R. F. MISSIO, E. C. SOUZA, C. N. FISCHER *et al.*, 2005 Satellyptus: analysis and database of microsatellites from ESTs of Eucalyptus. Genetics and Molecular Biology **28**.

COLLINS, J. R., R. M. STEPHENS, B. GOLD, B. LONG, M. DEAN *et al.*, 2003 An exhaustive DNA microsatellite map of the human genome using high performance computing. Genomics **82:** 10-19.

DE FONZO, V., E. BERSANI, F. ALUFFI-PENTINI, T. CASTRIGNANO and V. PARISI, 1998 Are only repeated triplets guilty? J Theor Biol **194:** 125-142.

DE RIDDER, C., D. KOURIE and B. WATSON, 2006 FireμSat: An algorithm to detect microsatellites in DNA, pp. in *Proceedings of the Prague Stringology Conference 2006*, Prague.

DELCHER, A. L., S. KASIF, R. D. FLEISCHMANN, J. PETERSON, O. WHITE *et al.*, 1999 Alignment of whole genomes. Nucleic Acids Res **27:** 2369-2376.

DELGRANGE, O., M. DAUCHET and E. RIVALS, 1999 Location of repetitive regions in sequences by optimizing a compression method. Pac Symp Biocomput**:** 254-265.

DELGRANGE, O., and E. RIVALS, 2004 STAR: an algorithm to Search for Tandem Approximate Repeats. Bioinformatics **20:** 2812-2820.

DOMANIC, N. O., and F. P. PREPARATA, 2007 A novel approach to the detection of genomic approximate tandem repeats in the Levenshtein metric. J Comput Biol **14:** 873-891.

EDGAR, R. C., and E. W. MYERS, 2005 PILER: identification and classification of genomic repeats. Bioinformatics **21 Suppl 1:** i152-158.

ELLEGREN, H., 2000 Heterogeneous mutation processes in human microsatellite DNA sequences. Nat Genet **24:** 400-402.

ELLEGREN, H., 2004 Microsatellites: simple sequences with complex evolution. Nat Rev Genet **5:** 435-445.

GIARDINE, B., C. RIEMER, R. C. HARDISON, R. BURHANS, L. ELNITSKI *et al.*, 2005 Galaxy: A platform for interactive large-scale genome analysis. Genome Res. **15:** 1451-1455.

GISH, W., 1996-2007 BLASTN 2.0 Academic Non-Commercial, pp. in *WUBLAST*. Washington University School of Medicine, St. Louis, Washington.

GROVER, A., V. AISHWARYA and P. C. SHARMA, 2007 Biased distribution of microsatellite motifs in the rice genome. Mol Genet Genomics **277:** 469-480.

GU, W., T. A. CASTOE, D. J. HEDGES, M. A. BATZER and D. D. POLLOCK, 2008 Identification of repeat structure in large genomes using repeat probability clouds. Anal Biochem **380:** 77-83.

GUR-ARIE, R., C. J. COHEN, Y. EITAN, L. SHELEF, E. M. HALLERMAN *et al.*, 2000 Simple sequence repeats in *Escherichia coli*: abundance, distribution, composition, and polymorphism. Genome Res **10:** 62-71.

HALL, T. A., 1999 BioEdit: a user-friendly biological sequence alignment editor and analysis program for Windows 95/98/NT. Nucl. Acids. Symp. Ser. **41:** 95-98.

HAUBOLD, B., and T. WIEHE, 2006 How repetitive are genomes? BMC Bioinformatics **7:** 541.

HAUTH, A. M., and D. A. JOSEPH, 2002 Beyond tandem repeats: complex pattern structures and distant regions of similarity. Bioinformatics **18 Suppl 1:** S31-37.

JENSEN, L. E., P. A. JAUERT and D. T. KIRKPATRICK, 2005 The large loop repair and mismatch repair pathways of *Saccharomyces cerevisiae* act on distinct substrates during meiosis. Genetics **170:** 1033-1043.

JURKA, J., 1998 Repeats in genomic DNA: mining and meaning. Curr Opin Struct Biol **8:** 333-337.

JURKA, J., J. WALICHIEWICZ and A. MILOSAVLJEVIC, 1992 Prototypic sequences for human repetitive DNA. J Mol Evol **35:** 286-291.

KANNAN, S. K., and E. W. MYERS, 1996 An algorithm for locating nonoverlapping regions of maximum alignment score. SIAM Journal on Computing **25:** 648-662.

KARACA, M., M. BILGEN, A. N. ONUS, A. G. INCE and S. Y. ELMASULU, 2005 Exact tandem repeats analyzer (E-TRA): a new program for DNA sequence mining. J Genet **84:** 49-54.

KAROLCHIK, D., R. BAERTSCH, M. DIEKHANS, T. S. FUREY, A. HINRICHS *et al.*, 2003 The UCSC Genome Browser Database. Nucl. Acids Res. **31:** 51-54.

KOFLER, R., C. SCHLÖTTERER and T. LELLEY, 2007a The SciRoKo 3.1 Manual, pp., Vienna, Austria.

KOFLER, R., C. SCHLÖTTERER and T. LELLEY, 2007b SciRoKo: A new tool for whole genome microsatellite search and investigation. Bioinformatics **23:** 1683-1685.

KOLPAKOV, R., G. BANA and G. KUCHEROV, 2003 mreps: efficient and flexible detection of tandem repeats in DNA. Nucl. Acids Res. **31:** 3672-3678.

KOLPAKOV, R. M., and G. KUCHEROV, 2001 Finding approximate repetitions under Hamming distance, pp. in *Proceedings of the 9th Annual European Symposium on Algorithms*. Springer-Verlag.

KOLPAKOV, R. M., and G. KUCHEROV, 2003 Finding approximate repetitions under Hamming distance. Theor. Comput. Sci. **303:** 135-156.

KOTA, R., R. K. VARSHNEY, T. THIEL, K. J. DEHMER and A. GRANER, 2001 Generation and comparison of EST-derived SSRs and SNPs in barley (*Hordeum vulgare* L.). Hereditas **135:** 145-151.

KURTZ, S., A. NARECHANIA, J. C. STEIN and D. WARE, 2008 A new method to compute K-mer frequencies and its application to annotate large repetitive plant genomes. BMC Genomics **9:** 517.

KURTZ, S., E. OHLEBUSCH, C. SCHLEIERMACHER, J. STOYE and R. GIEGERICH, 2000 Computation and visualization of degenerate repeats in complete genomes. Proc Int Conf Intell Syst Mol Biol **8:** 228-238.

KURTZ, S., and C. SCHLEIERMACHER, 1999 REPuter: fast computation of maximal repeats in complete genomes. Bioinformatics **15:** 426-427.

LA ROTA, M., R. KANTETY, J.-K. YU and M. SORRELLS, 2005 Nonrandom distribution and frequencies of genomic and EST-derived microsatellite markers in rice, wheat, and barley. BMC Genomics **6:** 23.

LANDAU, G. M., E. W. MYERS and J. P. SCHMIDT, 1998 Incremental String Comparison. SIAM J. Comput. **27:** 557-582.

LANDAU, G. M., J. P. SCHMIDT and D. SOKOL, 2001 An algorithm for approximate tandem repeats. J Comput Biol **8:** 1-18.

LANDAU, G. M., U. VISHKIN and R. NUSSINOV, 1987 An efficient string matching algorithm with K substitutions for nucleotide and amino acid sequences. J Theor Biol **126:** 483-490.

LEACH, R. W., and C. CLELAND, 1997 Tandyman, pp. Los Alamos National Laboratory, California, USA.

LECLERCQ, S., E. RIVALS and P. JARNE, 2007 Detecting microsatellites within genomes: significant variation among algorithms. BMC Bioinformatics **8:** 125.

LEFEBVRE, A., T. LECROQ, H. DAUCHEL and J. ALEXANDRE, 2002

FORRepeats: detects repeats on entire chromosomes and between genomes. Bioinformatics **19:** 319-326.

LEGENDRE, M., N. POCHET, T. PAK and K. J. VERSTREPEN, 2007 Sequence-based estimation of minisatellite and microsatellite repeat variability. Genome Res **17:** 1787-1796.

LI, R., J. YE, S. LI, J. WANG, Y. HAN *et al.*, 2005 ReAS: Recovery of ancestral sequences for transposable elements from the unassembled reads of a whole genome shotgun. PLoS Comput Biol **1:** e43.

MAIN, M. G., and R. J. LORENTZ, 1984 An O(*n* log *n*) algorithm for finding all repetitions in a string. Journal of Algorithms **5:** 422-432.

MAYER, C., 2007 Phobos Version 3.3.2. A tandem repeat search program, pp.

MERKEL, A., and N. GEMMELL, 2008 Detecting short tandem repeats from genome data: opening the software black box. Brief Bioinform **9:** 355-366.

MILOSAVLJEVIC, A., and J. JURKA, 1993 Discovering simple DNA sequences by the algorithmic significance method. Comput Appl Biosci **9:** 407-411.

MORGANTE, M., M. HANAFEY and W. POWELL, 2002 Microsatellites are preferentially associated with nonrepetitive DNA in plant genomes. Nat Genet **30:** 194-200.

MORGULIS, A., E. M. GERTZ, A. A. SCHAFFER and R. AGARWALA, 2006 WindowMasker: window-based masker for sequenced genomes. Bioinformatics **22:** 134-141.

MUDUNURI, S. B., and H. A. NAGARAJARAM, 2007 IMEx: Imperfect Microsatellite Extractor. Bioinformatics **23:** 1181-1187.

PALMIERI, D. A., V. M. NOVELLI, M. BASTIANEL and E. AL., 2007 Frequency and distribution of microsatellites from ESTs of citrus. Genetics and Molecular Biology **30**.

PARISI, V., V. DE FONZO and F. ALUFFI-PENTINI, 2003 STRING: finding tandem repeats in DNA sequences. Bioinformatics **19:** 1733-1738.

PRASAD, M. D., M. MUTHULAKSHMI, K. P. ARUNKUMAR, M. MADHU, V. B. SREENU *et al.*, 2005 SilkSatDb: a microsatellite database of the silkworm, *Bombyx mori*. Nucleic Acids Res **33:** D403-406.

PRICE, A. L., N. C. JONES and P. A. PEVZNER, 2005 *De novo* identification of repeat families in large genomes. Bioinformatics **21 Suppl 1:** i351-358.

RICHARD, G.-F., A. KERREST and B. DUJON, 2008 Comparative Genomics and Molecular Dynamics of DNA Repeats in Eukaryotes. Microbiol. Mol. Biol. Rev. **72:** 686-727.

RIGOUTSOS, I., and A. FLORATOS, 1998 Combinatorial pattern discovery in biological sequences: The TEIRESIAS algorithm. Bioinformatics **14:** 55-67.

RIVALS, E., O. DELGRANGE, J. P. DELAHAYE, M. DAUCHET, M. O. DELORME *et al.*, 1997 Detection of significant patterns by compression algorithms: the case of approximate tandem repeats in DNA sequences. Comput Appl Biosci **13:** 131-136.

ROZEN, S., and H. J. SKALETSKY, 1998 Primer3, pp. Whitehead Institute for Biomedical Research.

SAGOT, M. F., and E. W. MYERS, 1998 Identifying satellites and periodic repetitions in biological sequences. J Comput Biol **5:** 539-553.

SAHA, S., S. BRIDGES, Z. V. MAGBANUA and D. G. PETERSON, 2008 Empirical comparison of ab initio repeat finding programs. Nucleic Acids Res **36:** 2284-2294.

SCHMIDT, J. P., 1998 All highest scoring paths in weighted grid graphs and their application to finding all approximate repeats in strings. SIAM J. Comput. **27:** 972-992.

SHARMA, D., B. ISSAC, G. RAGHAVA and R. RAMASWAMY, 2004 Spectral Repeat Finder (SRF): identification of repetitive sequences using Fourier transformation. Bioinformatics **20:** 1405-1412.

SHARMA, P. C., A. GROVER and G. KAHL, 2007 Mining microsatellites in eukaryotic genomes. Trends Biotechnol.

SIM, J. S., K. PARK, C. S. ILIOPOULOS and W. F. SMYTH, 1999 Approximate Periods of Strings pp. 123-133 in *Combinatorial Pattern Matching*. Springer Berlin / Heidelberg.

SMIT, A., P. GREEN and R. HUBLEY, 1996-2007 RepeatMasker Open-3.0, pp.

SOKOL, D., G. BENSON and J. TOJEIRA, 2007 Tandem repeats over the edit distance. Bioinformatics **23:** e30-35.

SREENU, V. B., G. RANJITKUMAR, S. SWAMINATHAN, S. PRIYA, B. BOSE *et al.*, 2003 MICAS: a fully automated web server for microsatellite extraction and analysis from prokaryote and viral genomic sequences. Appl Bioinformatics **2:** 165-168.

STOLOVITZKY, G., Y. GAO, A. FLORATOS and I. RIGOUSTOS, 1999 Tandem-repeat detection using pattern discovery, with applications to the study of yeast satellites, pp. IBM T.J. Watson Research Center, Yorktown Heights, NY 10598.

TEMNYKH, S., G. DECLERCK, A. LUKASHOVA, L. LIPOVICH, S. CARTINHOUR *et al.*, 2001 Computational and experimental analysis of microsatellites in rice (*Oryza sativa* L.): frequency, length variation, transposon associations, and genetic marker potential. Genome Res **11:** 1441-1452.

THIEL, T., W. MICHALEK, R. K. VARSHNEY and A. GRANER, 2003 Exploiting EST databases for the development and characterization of gene-derived SSR-markers in barley (*Hordeum vulgare* L.). Theor Appl Genet **106:** 411-422.

THURSTON, M. I., and D. FIELD, 2005 Msatfinder: detection and characterisation of microsatellites, pp. CEH Oxford.

VOLFOVSKY, N., B. J. HAAS and S. L. SALZBERG, 2001 A clustering method for repeat analysis in DNA sequences. Genome Biol **2:** RESEARCH0027.

WEXLER, Y., Z. YAKHINI, Y. KASHI and D. GEIGER, 2005 Finding approximate tandem repeats in genomic sequences. J Comput Biol **12:** 928-942.

ZANE, L., L. BARGELLONI and T. PATARNELLO, 2002 Strategies for microsatellite isolation: a review. Mol Ecol **11:** 1-16.

CHAPTER III: Optimization of Approximate Microsatellite Searches

Abstract

In this chapter I continue the optimization of search parameters for microsatellite mining software focusing on the characterization of microsatellite imperfection structure. I compare the capacity of the programs TRF and SciRoKo, two of the most advantageous tandem repeat finders, to identify microsatellites and to extend through imperfections, screening all available parameter options. The main issues complicating tandem repeat identification within DNA sequences are imperfect motif conservation and complex pattern structures (mixtures of different motifs). However, it is essential to include imperfect microsatellites in whole genome scans because, as shown here, the majority of perfect microsatellite hits are included within longer imperfect versions of the same or different microsatellite loci. Moreover, the number of microsatellites decreases, while genome coverage increases, when imperfections and interruptions are allowed within repeat units. This is because several microsatellites are often found close to each other forming groups or clusters of 'adjacent microsatellites'. It is possible that such clusters of microsatellites did evolve from a single ancestral microsatellite. Therefore, the inclusion of imperfect microsatellites ('approximate repeats' extended through imperfections as long as a significant periodicity is satisfied) in studies of microsatellite distribution may be decisive when attempting to trace the evolution of microsatellite loci, particularly the intragenomic relationships among loci. The program TRF showed to be better suited than SciRoKo for extending microsatellite hits through imperfections, but it missed around 30 to 70% of the microsatellites contained in the test sequences, mostly short hits corresponding to short motifs. SciRoKo on the other hand, identified most perfect tandem repeats, although showing problems when microsatellite hits were adjacent to each other, and it extended relatively well through imperfections. Therefore, instead of choosing one of the programs, I decided to use both as complementary tools, in order to obtain the best possible representation of perfect and imperfect microsatellites within genomes.

3.1 Introduction

Throughout the analysis of tandem repeat finding algorithms in Chapter II I showed that the parameter settings used in a search run, which are usually program-specific, can have a strong influence on the search outcome. These differences have also been studied by Leclercq *et al.* (2007) and Merkel and Gemmell (2008) and, although these studies had a few drawbacks (see discussion in Chapter II, page 90), their conclusions about the parameter differences were substantiated. Therefore, in this Chapter I present detailed comparisons of output datasets produced by the programs TRF (BENSON 1999) and SciRoKo (KOFLER *et al.* 2007b), which were shown in Chapter II to offer the most useful features for comparative genomics, in order to choose the parameter settings which yield the best representation of microsatellites in the query sequences.

For the optimization of microsatellite searches, the search parameters need to be set so that they represent as close as possible the biological properties that are likely to be involved in microsatellite dynamics. Nevertheless, it is striking to observe that microsatellites are usually referred to, in the general literature as well as in earlier bioinformatic papers, as single, independent and well defined units within DNA sequences (for example see KARACA *et al.* 2005). Microsatellite searches and analyses usually focus on characterizing individual short perfect tandem repeats, counting these independently, regardless of their closeness to other similar or different repeated motifs (see CASACUBERTA *et al.* 2000; COLLINS *et al.* 2003; FUJIMORI *et al.* 2003; GUO *et al.* 2007). However, it is known that microsatellites are not only perfect reiterations of a motif, but that these can acquire a range of imperfections and interruptions to the repeat (TAUTZ *et al.* 1986). These imperfections can be base substitutions, insertions or deletions, and recombination exchanges. Tandem arrangements of closely related motifs are also often found (LEVINSON and GUTMAN 1987).

When visualizing the microsatellite positions from the output of a perfect microsatellite search, the hits seem to occur in clusters, and oftentimes two or more hits could be better interpreted as a single one containing interruptions. This can be illustrated by highlighting single motifs across a DNA sequence, as shown in **figures 3.1, 3.2** and **3.3**. The first thing to observe in the figures is that there are regions composed of perfect repeats which can eventually be interrupted by insertions, deletions, and substitutions, seemingly dividing one microsatellite into two or more pieces. Using an approximate tandem repeat finder, like TRF and SciRoKo, with relaxed parameters with respect to the penalties for interruptions, whole groups of dense same-motif occurrences like the ones depicted could be reported as one

long imperfect microsatellite, instead of the 2, 13 and 12 microsatellite hits that a perfect microsatellite finder would report (respectively for **figures 3.1, 3.2, 3.3)**. A second point is that these tandem repeat groups can also be interspersed with tandems of more than one motif, as is the case in **figure 3.1**. Depending on the parameter settings used for the search, these different motifs can be detected as interruptions, and be therefore extended through so that a longer imperfect microsatellite is reported, or they can be reported as separated microsatellites with different motifs. The occurrence of a mix of different microsatellite motifs in an adjacent fashion is referred in the literature as 'compound microsatellites' (CHAMBERS and MACAVOY 2000).

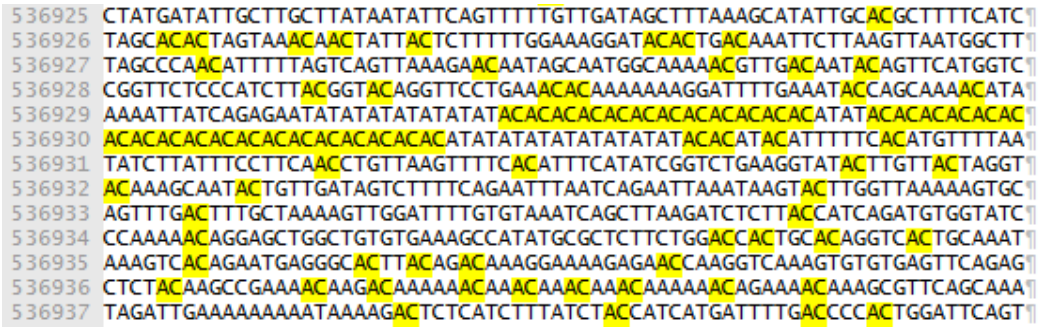

Figure 3.1: Illustration of a relatively well defined AC repeat in human chromosome 22 visualized with the text editor Vim (AC motif highlighted in yellow).

Figure 3.2: Illustration of a long imperfect TGGA microsatellite in the human chromosome 22 (TGGA motif highlighted in yellow).

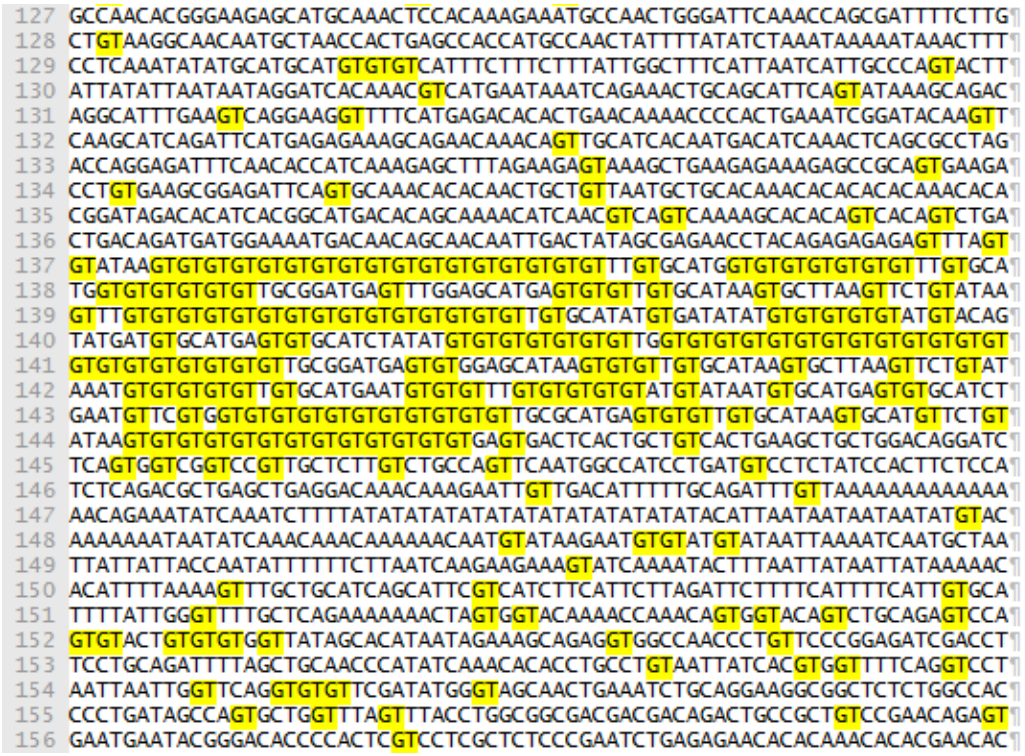

Figure 3.3: Illustration of a dispersed group of GT tandem repeats in the chromosome 1 of zebrafish (*Danio rerio*) visualized with the text editor Vim (AC motif highlighted in yellow).

Imperfections and interruptions within a microsatellite tract are believed to stabilize it (i.e. to make it less prone to mutation by replication slippage) by decreasing the consecutive tandem repeats that can undergo replication slippage. This hypothesis is supported by a markov chain model developed on yeast di-, tri- and tetranucleotide distributions by Kruglyak (2000), and several *in vivo* studies (BRANDSTROM and ELLEGREN 2008; SYMONDS and LLOYD 2003; VAN OPPEN *et al.* 2000; VAN TREUREN *et al.* 1997). It is also assumed that microsatellites degenerate through the accumulation of base substitutions (CHAMBERS and MACAVOY 2000), because the interrupting bases divide microsatellites into smaller repeat runs with different mutational characteristics, in general decreasing the mutation rate (BOYER *et al.* 2002). However, complex microsatellites, containing mixtures of different motifs, which initially are also formed by interrupted microsatellites, have been shown to reach very high polymorphism (DOXIADIS *et al.* 2007; UDAR *et al.* 1999; WOOD and SCHERTZER 1992). Microsatellite interruptions can also eventually be removed by the same replication slippage process that is involved in microsatellite mutation (HARR *et al.* 2000). Evidently, interruptions and imperfections are part of the evolution of microsatellites, and need therefore to be analyzed in parallel with these in order to be included in analyses of microsatellite abundance and distribution if we want to understand microsatellite dynamics.

A very small number of microsatellite distribution studies have included imperfect microsatellites in their searches and, among these, the amount of imperfection allowed within the microsatellites was always very limited. Crollius *et al.* (2000), Morgante *et al.* (2002), and Edwards *et al.* (1998) allow 1 mismatch per 12 nt or a minimum of 85% perfection. Katti *et al.* (2001) allowed one mismatch for every 10 nt, Malpertuy *et al.* (2003) considered microsatellites containing at least 5 uninterrupted repeats, and Yeramian *et al.* (1999) opted for a minimum of 80% perfection as allowed by the TRF mismatch probability (but they used very stringent mismatch and indel penalties, setting the respective parameter values at their maximum: 7). Moreover, there is no representative study which attempts to quantify or characterize the degree of imperfection in microsatellites in different genomes. However, a recent study by Kofler *et al.* (2008), the authors of the program SciRoKo, presented an analysis of microsatellite clustering where microsatellites separated by 10 or less nucleotides in the sequence were quantified. They concluded that about 4 to 25% of all microsatellites in 8 mammalian genomes analyzed could be categorized as compound microsatellites with the described characteristics.

The presence of imperfections and interruptions within microsatellite sequences, as well as other complex pattern structures like mixtures of different motifs and arrays with fuzzy motif transitions, complicate the task of finding imperfect microsatellites. This is probably the reason for the scarcity of studies on imperfect microsatellites. However, with the constant development of approximate tandem repeat finders, as evidenced in Chapter II, new possibilities for this kind of analysis become available. The programs TRF (BENSON 1999) and SciRoKo (KOFLER *et al.* 2007b) have an extensive set of parameters to control for the proportion of imperfections and interruptions within microsatellite hits. I therefore focused during the comparisons among TRF and SciRoKo search results, on the capacity of each of the programs to extend microsatellite hits efficiently through imperfections and interruptions, as long as the global periodicity remains significant, in order to analyze the extent to which the microsatellites in a genome are imperfect and/or part of more complex repeat structures.

3.2 Methodology

3.2.1 Program output comparison: TRF *vs* SciRoKo

The characterization of search parameter effects for the programs SciRoKo (KOFLER *et al.* 2007b) and TRF (BENSON 1999), and the optimization of these searches for the analysis of approximate microsatellites for comparative genomics purposes, were carried out in this Chapter using the same test sequences from Chapter II (**table 2.4**, page 72).

A custom perfect tandem repeat finder, IrSa, was used to obtain a reference set of microsatellite hits for each test sequence except the human chromosome 22. The program IrSa was developed in collaboration with Carsten Horn (Gloon Systems) to find and quantify exhaustively perfect tandem repeats for each microsatellite motif independently (see methodology in Chapter IV for the program's details). The comparison of search results between the tested programs and the IrSa reference datasets were carried out using the interval manipulation tools at the Galaxy webpage (http://g2.bx.psu.edu, GIARDINE *et al.* 2005). A closer observation and comparison of results from the shorter test sequences was performed with the aid of the sequence visualization tool Bioedit ver 7.0.5.3 (HALL 1999). During the visual comparisons, the ability of each program, and of each parameter combination within these, to extend through imperfections was tabulated as categorical variables on Excel files

TRF

In the program TRF, the available parameters are given to the program in the following order: match points, mismatch penalty (absolute value), indel penalty (absolute value), match probability, indel probability, minscore, and maximum motif length (i.e. 2 3 5 80 10 30 6). The mismatch and indel penalties, which are useful to fine-tune the extension of the search through imperfections and interruptions, were tested first, keeping the match points, match probability, indel probability, and minscore constant with the values 2, 80, 10, and 30, respectively. The maximum motif length was set to 6 nt. The values tested for the mismatch and indel penalties ranged from 3 to 7 points. Selected value combinations for the match and indel penalties were then tested with the values of 75 and 20 for the mismatch and indel probabilities, and with the minscore value ranging from 2 to 30.

SciRoKo

The command line version of the program SciRoKo, SciRoKoCo, was used for all the tests. It has all the basic search modes and options, but does not include the SSR-statistics module. Even though I used mainly the command line version, I will refer to it as SciRoKo to avoid confusion. SciRoKo has five search modes; three for perfect microsatellites and two modes for imperfect microsatellites. The perfect search modes differ in the way the minimum microsatellite length for the search is set: the perfect repeat mode (**-pr**) in number of repetitions, the perfect length mode (**-pl**) in number of nt, and for the the misa mode (**–misa**) the minimum number of repetitions is set individually for each motif length. The perfect search modes were compared with IrSa results with a minimum length thresholds ranging, from 3 to 16 repetitions for the pr mode, 3 to 20 nt for the pl mode, and varying only the minimum length threshold for mononucleotides and dinucleotides from 3 to 8 repeats in the case of the misa mode.

The two imperfect search modes of SciRoKo differ in the way the scores are calculated. The score for the "mismatch fixed penalty" mode (**-mmfp**) is calculated with the formula: number of hits - number of mismatches × mismatch penalty, and the score for the "mismatch variable penalty" mode (**-mmvp**) is calculated with another formula: number of hits - number of mismatches × (mismatch penalty × motif length). Each mode was tested by varying the mismatch penalty (**-p**) from 1 to 5, the seed length (**-seedl**) from 2 to 8, and the score (**-s**) from 3 to 16. The seed repeats (**-seedr**) and maximum mismatches at once (**-mmao**) were first kept constant at a value of 3 each. In the final testing runs with selected -seeld and –p values, the -mmao parameter was tested with values from 3 to 16.

The results from all the tests were graphed to obtain microsatellite number and coverage distributions, to observe the variation in the magnitude of the datasets by changing the search characteristics.

Exact quantitative comparative analyses between the resulting datasets from TRF, SciRoKo, and IrSa, with equivalent parameter settings, were performed mainly using the tools for operation on genomic intervals at the Galaxy web page http://main.g2.bx.psu.edu/ (GIARDINE *et al.* 2005). The program IrSa reports any tandem repeat in the input sequence, regardless of overlaps with other hits, and it finds exhaustively all tandem repeats of the motif lengths chosen in the input. The program TRF also reports redundant hits, and

therefore the results from TRF and IrSa were first merged within each dataset (Merge option in Galaxy). The merging process replaces overlapping intervals with a single interval spanning all hits in the overlapping group. The lowest and highest coordinate values from the group are kept for the new interval. The proportions of IrSa hits identified by each program with each parameter set tested were calculated using both, numbers and coverage of the microsatellites (as measured by the Base Coverage option in Galaxy).

The qualitative assessment of results was performed based on the quantitative comparisons in Galaxy, with subscequent visualtisation of microsatellite hits with Bioedit (HALL 1999). Differences in the length of microsatellite hits among datasets were first identified with the substract tool in Galaxy (non-overlapping pieces of intervals). In most cases, merged datasets were used, and during the merging process in Galaxy all information except start and end coordinates of the microsatellite hits gets lost. Therefore, the coordinates were used to build custom tracks to be loaded into the UCSC Genome Browser (http://www.genome.ucsc.edu/) to retrieve the sequences of the microsatellites. Start and end positions, and the respective sequences, were then compared in Excel spreadsheets to assess the capacity of TRF and SciRoKo to extend hits through imperfections by visual analyses of adjacent tandem repeats in the IrSa datasets.

3.2.2 Organization and analysis of program output

For the subsequent large scale analyses of genomic sequences, the tasks of filtering, merging, and quantifying numbers and coverage in microsatellite datasets were automated in a series of scripts written in collaboration with Lisha Naduvilezhath, an exchange student from the Wolfgang Goethe University in Germany, in Visual Basics for Excel and Java languages. This was necessary because the tools from Galaxy can not be automated online, and it was not possible to install a Galaxy server at the University of Canterbury Supercomputer (the Galaxy Python scripts would have required a substantial amount of fine-tuning because these were under constant development; personal communication from Vladimir Mencl from the BlueFern team http://www.bluefern.canterbury.ac.nz/, who tested the scripts). The programs developed are listed below:

- DeleteFirstLines.java: Script for the elimination of comment lines from the output of search programs.
- MsatFilter.java: Script to filter out too short and too imperfect microsatellite hits from the TRF output by using the information given in the "period size" (equivalent to

motif length), "repeat number", and "percent matches" columns (columns 3,4, 6, respectively).

- Merge.java: Script to merge overlapping rows into one. It keeps only the minimum start and maximum end coordinates from among the overlapping hits (**figure 3.4**), and produces an output file containing only the sequence name, start, and end coordinates. The minimum overlap length for rows to be merged can be specified in the code; the default is 3 nt. This tool can be used for quick redundancy elimination, and as a pre-treatment to join output datasets from different programs.

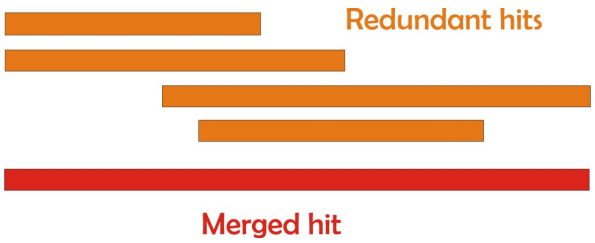

Figure 3.4: Illustration of merging redundant hits into one

- TRFredundancyEliminator.java: Script designed to eliminate redundant hits from TRF output datasets, while keeping the detailed information in the other columns describing the microsatellite characteristics. It reduces output rows with overlapping start or end coordinates to one, given that the overlap is higher than a value specified. The overlapping rows are replaced with a new row keeping as coordinates the minimum start and maximum end coordinates among the overlapping ones. The rest of the information contained in the other columns is kept only for the longest hit. The motif information of all overlapping hits is concatenated in the motif column. The resulting dataset retains all columns from the original output dataset.
- MsatDensity.java: This script calculates the coverage and percent of microsatellties for chromosomes divided into intervals 100000 nt intervals (the length of the intervals can be modified), or as a whole. It takes only merged datasets as input.

All programs were controlled and run in BATCH using a Cygwin (http://www.cygwin.com/) console. The operations performed with these scripts are detailed in the results section in the order in which they were applied, and citing each script.

3.3 Results and Discussion

Here I present an in-depth comparison of the microsatellite search capacities of the programs TRF and SciRoKo, and the optimization of search parameters for the application of these programs in comparative microsatellite analyses between different genomes. The search strategies from TRF and SciRoKo are essentially different as evidenced by their distinct sets of search parameters shown in **table 3.1** (parameters which modify similar characteristics in the searches are shown in the same row). Numerical values need to be assigned to all these parameters, and the effect of each of them on the search results was characterized by varying single parameters at a time though a range of values.

Table 3.1: Comparison of search parameter options for TRF and SciRoKo

Main search parameters for the definition of microsatellites	Tandem Repeat Finder (TRF)	SciRoKo (SciRoKoCo)
min unit length	na	na (default 1)
max unit length	Maxperiod, 7th value	na (default 6)
points for a match	1st value	na
mismatch penalty	2nd value	-p
indel penalty	3rd value	na
min score	6th value	-s
min length	minscore/match points	-l -seedl -seedr
min number of repetitions	na	-r, -m (for MISA mode)
imperfection	implied in match and indel probabilities (PM and PI): 4th and 5th value	-mmao (maximum mismatches at once)

I will first discuss the individual characterization of parameter effects for each program, and then compare the microsatellite identification capacity of both programs.

3.3.1 TRF

TRF is a repeat finder specialized for finding approximate tandem repeats with repeat units up to 2000 nt in length. Upon its release in 1999, the program was only suitable for the analysis of relatively small DNA sequences, up to 700 kb (BENSON 1999). The capacity of the

program seemingly increased in future releases, as Castelo *et al.* (2002) pointed out that the upper sequence length limit for processing with TRF was 5Mb. Moreover, since the release of version 3.01 in March 2002, the sequence length limit was overcome completely (Yevgeniy Gelfand, personal communication). Nevertheless Sharma *et al.* (SHARMA *et al.* 2007) mistakenly assumed that TRF (the current version is 4.00) is still not able to process sequences longer than 5 Mb.

To date, TRF appears to be the most frequently used tandem repeat finder (for examples see AMES *et al.* 2008; for examples see KAROLCHIK *et al.* 2003; WARBURTON *et al.* 2008; YERAMIAN and BUC 1999), and it was used to construct at least five of the main microsatellite databases (see **table 2.2**).

Imperfections

The detection of imperfections with the program TRF is based on a probabilistic model, and therefore the amount of imperfection allowed in a tandem repeat hit is relative to its total length. These probabilistic values are the mismatch[1] probability and the indel probability, which have the default values of 80 and 10, respectively. With these values, an average of 80% mismatches and 10% indels would be allowed among the repetitions of a tandem repeat. These probabilities are also referred to by the author as a type of "extremal bound" (BENSON 1999). However, the maximum imperfection contained within search hits can not be restricted with these values, because the imperfection content of reported hits will also depend on the mismatch and indel penalties and the minimum score value (minscore). In practice, with the most restrictive parameters, 2 7 7 80 10 x 6, percent matches as low as 63% were observed, and with very relaxed parameters 2 3 6 75 20 x 6, percent matches as low as 37% were observed.

The version of TRF I used throughout this research was 4.00. This version only allows setting the probability values to 80 and 10, or 75 and 20, for the match and indel probabilities, respectively. These values can not be set to 100 and 0 which means that TRF can not search exclusively for perfect repeats; even with the most stringent settings 2 7 7 80 10 x 6, there will still be small interruptions within the hits. For this reason, a good perfect repeat finder using equivalent restrictions for the minimum length of the hits should usually find more microsatellite hits than TRF.

[1] Substitutions are systematically referred to as mismatches in the TRF documentation. This contrasts with the terminology used in the SciRoKo documentation, where the term 'mismatch' is used to refer to both, substitutions and indels.

The second and third parameter values for TRF, corresponding to the mismatch and indel penalties, offer relatively few modification possibilities. Based on output hit range comparisons with the empirical microsatellite sets for each test sequence, I chose a mismatch penalty value of 3 and an indel penalty of 6 from all combinations of mismatch and indel penalty values from 3 to 7,. The mismatch value needs to be low to allow hit extension through long imperfect microsatellites, and the indel penalty has a relatively high value in an attempt to avoid that adjacent microsatellites with different motifs get reported as a single hit (hit merging). However, tuning the indel penalty has only a general effect on hit merging by controlling the amount of interruptions within a microsatellite. The penalties and score calculations are the same irrespective of motif length and therefore no distinction will be made among motifs. Regardless, the overall coverage and identification of microsatellite regions was most similar to the empirical datasets with the set of parameters 2 3 6 for the score points, and 75 20 for the probability values.

Authors who used TRF for microsatellite searches usually used the most restrictive parameters, setting both mismatch and indel penalties to the maximum value, 7 (CROLLIUS *et al.* 2000; YERAMIAN and BUC 1999). The UCSC Genome Browser also generates microsatellite tracks for all genomes with mismatch and indel penalties of 7 (i.e. 2 7 7 80 10 50 2000, personal communication from the UCSC mailing list, genome@lists.soe.ucsc.edu, http://www.genome.ucsc.edu/). The TRbase (http://trbase.exeter.ac.uk/advtr.html) and InSatDb (ARCHAK *et al.* 2007) use some lower parameter combination values: TRbase with 2 7 7 and 2 5 5 for match, mismatch, and indel penalties, and the InSatDb applied the parameter sets 2 3 5 80 10 45 and 2 5 7 80 10 30 to build microsatellite databases (see Table 2.2 in Chapter II).

Redundancy

TRF reports redundant hits in its output, and number of these hits varies from 2 to 5 per microsatellite-like region. Redundant hits are microsatellite hits with different repeat motifs whose coordinates overlap to a significant extent. Redundancy is a problem intrinsic to the search of tandem repeats, because tandem repeats can often be interpreted as repeats of more than one motif. The redundancy due to equivalent motifs, like CA=AC, CAT=ATC=TCA, GAAG=AAGG=AGGA=GGAA, etc, is the simplest kind of redundancy to deal with, because unambiguously only one of the motifs can represent all other motifs, and the one attaining the longest hit and/or the highest score is usually chosen. Therefore, this kind of redundancy is not reported by TRF nor the other programs tested in Chapter II. The

problem arises when the same sequence segment can be reported as more than one kind of motif, as shown in **table 3.2**. This is very often the case in approximate tandem repeats, when interruptions become periodic, probably due to the initial reproduction of one interruption by replication slippage, or when the motif of a long imperfect microsatellite transforms gradually into another motif. This kind of redundancy due to the detection of different motif lengths or non-equivalent motifs of the same length is more of a problem. The reason for this is two-fold: first, it is necessary to decide if the overlapping hits are truly redundant, referring to the same microsatellite, or if they correspond to different adjacent or slightly overlapping microsatellites, and second, the best motif representing the tandem repeat needs to be chosen. To exemplify the first problem, in **table 3.2**, groups 1, 4, and 6, represent truly redundant microsatellite hits, where choosing the hit with the best score can be considered a good solution. However, groups 2, 3, and 5 are heterogeneous groups, with some redundant hits corresponding to the same microsatellite, but hits overlapping by few nucleotides could be considered separate microsatellites.

After deciding that a group of redundant hits represent the same microsatellite, the most representative motif for the group needs to be chosen. This is usually circumvented by keeping the hit with the shortest motif to represent the repetitive region (AMES *et al.* 2008; CROLLIUS *et al.* 2000). However, sometimes longer motifs could attain higher lengths and alignment scores (BENSON 1999), because they can be extended more efficiently through imperfections. Benson (1999) addresses this problem by reporting the three hits with the best alignment scores from the alignment with a consensus sequence which takes place during the analysis component of the program run. In this way, the choice of the most suitable interpretation of redundant results is left to the user.

For the various reasons exposed here, the elimination of redundancy from output datasets of microsatellite searches is a problem which can not be unambiguously resolved by computer programs, at least not before knowing more about the interaction and relationship between adjacent microsatellites within genomes. Nevertheless, and contrary to the affirmation of Leclercq *et al.* (2007), the logic usually applied in algorithmic design, that a nucleotide can only belong to one microsatellite hit, is not biologically justified. If a nucleotide can form part of two different microsatellite hits, it could participate in mutations from any of both overlapping microsatellites.

Table 3.2: Examples of redundant hits reported by the program TRF

	Start	End	# repeats	Overlap nt	Motif	Sequence
1	3097718	3097750	5.5	38	TCTGTC	TCTGCCTGTGTCTCTGTCTCTCTGTATCTCT
	3097712	3097751	20		TC	TCTCCTTCTGCCTGTGTCTCTGTCTCTCTGTATCTCTC
2	14631058	14631116	9.7	56	TGTTTT	TGTTTTTTTTTTTTGTTTGTTTGTTTTTGGGTTTTTTTTTTTTTTTTGTTGTTGTTGTT
	14631060	14631101	42	1	T	TTTTTTTTTTTGTTTGTTTGTTTTTGGGTTTTTTTTTTTTTTT
	14631100	14631119	6.7		TTG	TTGTTTGTTGTTGTTGTTGTT
3	24463022	24463051	7	24	CTTT	CTTTCTTTTTTTTCTTTTTCTTTTTCTTTCTTT
	24463027	24463073	47	43	T	TTTTTTTCTTTTTCTTTTTCTTTTTCTTTATTTTTATTTTTATTTTTTTTT
	24463030	24463051	4.6	2	TTTTC	TTTTCTTTTTCTTTTTCTTTCTTT
	24463049	24463073	4.3		TTTATT	TTTATTTTTTATTTTTTATTTTTTTTT
4	6879551	6879586	6	31	ACAGAG	ACACAGAGAGAGAGAGGCAGAGACATAGGCAGAG
	6879555	6879591	18.5	32	AG	AGAGAGAGAGAGAGGCAGAGACATAGGCAGAGGGAGA
	6879559	6879596	6.3		AGAGGC	AGAGAGAGAGGCAGAGACATAGGCAGAGGGAGAAGCAG
5	14781992	14782037	15.3	45	GAG	GAGGAGGAGGAAGAGGAGGAAGAGGAGGAGGAGGAAGAGG
	14781992	14782055	10.3	10	GAGGAA	GAGGAGGAGGAAGAGGAGGAAGAGGAGGAGGAGGAAGAGGGGAAGAGGGAAGAGGAAGA
	14782045	14782079	11.3		GAA	GAAGAGGAAGAAGATGA...
6	772956	773004	8.2	48	GGGACG	GGGGCGGGGCCGGGAGCCGGGAAGGGGAGGGGGACGGGGGAGGGAGGG
	772956	773004	9.6		GGGGA	GGGGCGGGGCCGGGAGCCGGGAAGGGGAGGGGGACGGGGGAGGGAGGG
7	659979	659998	3.3	15	CGGTGC	CGATGCCCCGAGCCGGTGCCG
	659983	660115	26.4	120	GCCCG	GCCCGAGCCGGTGCCGCCGCCCCCGCGCCGCGCAGCCCCGGCCGCCAGGCCTTAGCCCGCCCCGGC...
	659995	660112	20.7	116	GCCCCC	GCCGCCGCCCCCGCGCCGCGCAGCCCCGGCCGCCAGGCCTTAGCCCGCCCCGGC...
	659996	660006	11	7	C	CCGCCGCCCCC
	659999	660135	22.5	135	CCGCAC	CCGCCCCCGCGCCGCAGCCCCGGCCGCCAGGCCTTAGCCCGCCCCGGCC...
	659999	660135	8.5		CG	CGCCCCCGCGCCGCGC

A redundancy filter, TRFredundancyEliminator.java, was designed to filter out redundant hits from TRF results, while keeping the detailed information given by TRF on the characteristics of the microsatellite. The program does not choose a motif to represent the redundant hits. Instead, it concatenates all motifs separated by dashes in the motif column. The coordinates to represent the redundant hits span all the hits by choosing the minimum start and maximum end position from the single hit coordinates. The decision of how many nucleotides need to overlap in order for two hits to be considered redundant is left to the user. When running TRFredundancyEliminator.java on TRF search results produced with different parameters, setting the maximum overlap length to 3 nt. I observed that the percentage of redundancy produced by TRF increases when decreasing the stringency of the parameters, and also when reducing the minimum score (minscore). This is because the longer and more imperfect the microsatellites reported, the more overlapping hits with different non-equivalent motifs can be found and reported. Mixtures of different motifs within a microsatellite, and fuzzy motif transitions, which are reported in higher quantity at lower stringency, tend to produce an increase in the overall number of redundant hits in the output.

Minimum length threshold: minscore

The minimum length threshold for TRF searches is set indirectly using the minscore value; The minscore value divided by the match value gives the minimum length of tandem repeats that will be reported. Usually the match value is set to 2, thus the minimum length for microsatellite hits would be half the minscore. This single value is used to filter all motifs searched for. Therefore, choosing a minscore value is a trade-off between setting the threshold high enough to avoid reporting too short tetra-, penta- and hexanucleotide repeats, and low enough so that the shorter motifs get reported satisfactorily. In case of requiring relatively short mono-, di-, and trinucleotides, it would be best to run two separate searches with maxperiods (maximum motif length) of 3 and 6. However, then the hits for mono-, di- and trinucleotides have to be filtered out from the second dataset because there is no option in TRF to set the minimum motif length to search for.

The lower limit for mono- and dinucleotide microsatellite length is restricted in TRF, by the length of the *k-tuple* used for the search (3 nt for motifs from 1 to 29 nt in length) and by the methods of comparison among *k-tuples* (BENSON 1999). Therefore, the program will report mononucleotides with a minimum size of 4 repeats even when minscore values

smaller than 10 are used. A minimum of 5 repeats will be reported more frequently. For dinucleotides, this minimum is 3 repeats, and for all other motifs the minimum is 2 repeats.

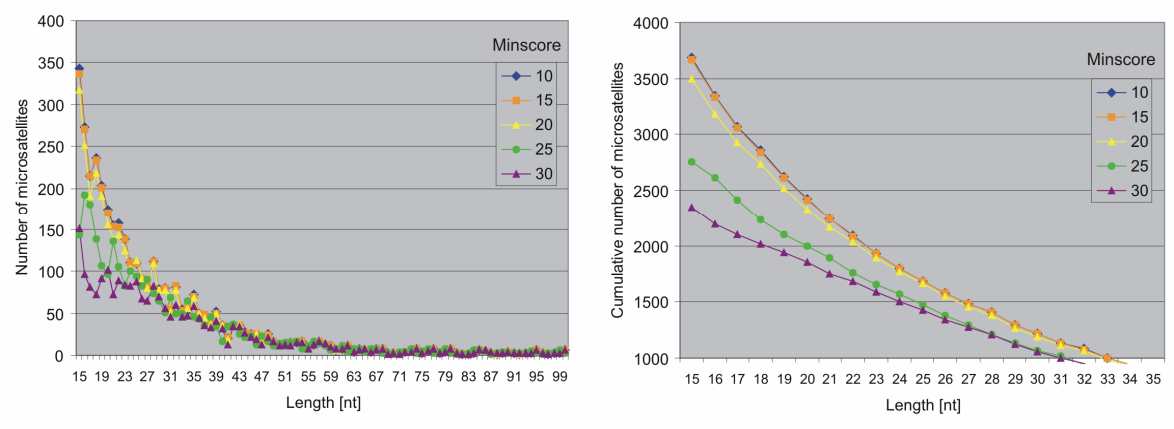

Figure 3.5: Microsatellite number and coverage distributions on the chromosome 1 of *Plasmodium falciparum* in relation to the length of microsatellite hits obtained with increasing minscore values (the other parameters used were 2 7 7 80 10 x 6, redundant hits were filtered out). The minscore value controls indirectly the minimum length of hits: minimum length=minscore/match value. The match value used for all searches was 2, so that the minimum length detected corresponds to half the minscore. As can be expected, the number of microsatellites reported increases exponentially when reducing the minscore value. However, this increase in numbers is not only due to smaller hits detected, but also to an increase in hits with longer lengths. This is evidenced by the change in the curves for microsatellite hits longer than 15 nt, which should not be affected by the minimum length constraint.

The minscore value also affects the TRF search in an unexpected way, probably due to its influence on the heuristics of the algorithm: The number of microsatellite hits, and the coverage of these, increase in an exponential fashion when reducing the minscore value. This exponential increase can, however, also be observed for longer microsatellites which are not directly affected by the minimum length constraint. For example, in **figure 3.5**, the number and coverage of microsatellite hits longer than 15 nt (which is the lower minimum microsatellite length allowed by the highest minscore used, 30) are plotted against a range of minscore values. A similar effect was also observed by Leclercq *et al.* (2007), by which lower minscore values produced more complete search results, albeit with an increase in false-positive hits (too short or too degraded hits). This increase in hits is also due to an increase in redundant hits because lower thresholds allow more redundant hits to be reported. However, the effect is proportionally the same after filtering out redundant hits. The datasets depicted in **figure 3.5** were previously filtered for redundancy, and therefore all additional hits observed when reducing the minscore value represent valid microsatellite

hits. Based on these observations, it would make sense to perform TRF searches at the lowest possible minscore values, and to use an additional filter for eliminating false positives afterwards

Figure 3.6 shows the microsatellite number and coverage distributions with respect to varying minscore for the test sequence plas1.fa. By comparison of the curves for the same parameters measured in number of microsatellites and in coverage, less stringent parameters with values of 2 7 7 for match points, and mismatch and indel penalties respectively, report a higher number of microsatellite hits, but these are shorter in comparison to the ones reported by the 2 3 6 combinations, because the overall coverage values for 2 7 7 combinations are lower. In all sequences tested, the minscore value reaches a plateau at values lower than 8. Moreover, the difference in microsatellites reported between minscores 8 and 10 is mainly due to reporting of mononucleotides with four repetitions with the lower minscore. Therefore, a minscore value of 10 was chosen for further TRF searches, and a java script was written to filter out false positive microsatellite hits based on the number of repetitions and the percentage of matches reported by TRF.

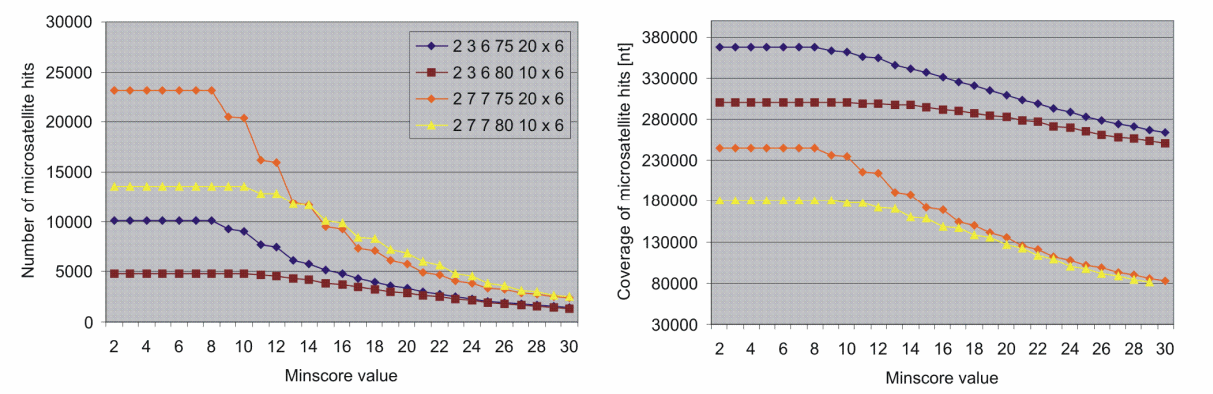

Figure 3.6: Microsatellite number and coverage distributions generated with TRF on the chromosome 1 of *Plasmodium falciparum*. The parameter settings for each curve are shown in the series in increasing order of stringency from top to bottom (the same parameters are shown in both graphs). As can be observed by the change in order of the curves between the number of repeats scale and the coverage scale, microsatellites reported by TRF get longer when more relaxed parameter settings are used, therefore reducing in numbers, but increasing in coverage values. Additionally, TRF detections reach a plateau zone at minscore values less than 8.

Microsatellite output datasets obtained with the parameter combinations 2 7 7 80 10 10 6 (most stringent set, TRF277) and 2 3 6 75 20 10 6 (very relaxed settings, TRF236) were compared in Galaxy with the IrSa empirical dataset, to verify the proportion of perfect

tandem repeats captured within the imperfect microsatellite hits generated by TRF. The TRF datasets were filtered with MsatFilter.java using the same thresholds used for the IrSa search, in this case 5, 3, 3, 3, 3, and 3 repeats for mono- to hexanucleotides, respectively, and all datasets were merged before the comparisons. **Table 3.3** shows that TRF missed high proportions of the perfect hits, and this proportion varied among test sequences. With stringent parameter settings TRF seems to miss more of the perfect hits than with stringent parameter settings, although the inverse scenario was observed in plas1.fa. The proportion of missed repeats may depend on the degree of imperfection and clustering of microsatellites within the query sequences. The sequence plas1.fa had the highest AT content among the test sequences (~80%), and the *Plasmodium* genome also contains the highest percentage coverage of microsatellites, as reported in Chapter IV of this document.

Table 3.3: Proportion of missing perfect microsatellites during TRF runs with two parameter sets (TRF236=2 3 6 75 20 10 6, TRF277=2 7 7 80 10 10 6), both individually, and after concatenating and merging both sets together. The TRF datasets were processed to filter out hits with less than 60% matches, and smaller than 5, 3, 3, 3, 3, and 3 repeats for mono- to hexanucleotides, respectively, to make them comparable to the empirical perfect microsatellite datasets based on the same minimum thresholds. Redundant hits were also filtered out previous to the comparisons.

Test sequence	Parameters	Proportion of missing perfect hits	Percentage coverage of missing hits
danio.fa	TRF236	33.48	16.39
	TRF277	40.18	18.30
	TRF236+TRF277	43.75	19.57
NC_009337.fa	TRF236	57.04	55.65
	TRF277	74.11	67.65
	TRF236+TRF277	48.93	44.42
plas1.fa	TRF236	47.90	43.17
	TRF277	38.61	22.28
	TRF236+TRF277	23.90	14.03
yeast1.fa	TRF236	58.54	51.75
	TRF277	67.29	54.81
	TRF236+TRF277	48.10	38.87

There were also substantial differences among the hits detected by TRF at low and high stringencies: The TRF236 parameter set (parameters 2 3 6 75 20 10 6) reported about 40 to 75% microsatellite hits that TRF277 (parameters 2 7 7 80 10 10 6) does not detect, while about 30% of the TRF277 output was not reported by TRF236. For this reason I decided to concatenate and merge the TRF results from both parameter sets to produce more representative datasets. The proportion of microsatellite hits missed gets reduced accordingly when using joint datasets from TRF236 and 277, but as seen in **table 3.3**, the proportion of missing hits is still high and, apparently, this proportion may depend on the sequence characteristics as well as the microsatellite characteristics (i.e. abundance and

imperfection) in the query sequence (see **table 3.4** to compare the characteristics of the test sequences with the coverage of missed hits).

The minimum length thresholds used in the comparisons with the empirical datasets were 5, 3, 3, 3, 3, and 3 repeats for mono- to hexanucleotides, respectively. These thresholds may be considered too low for usual microsatellite searches. When using higher thresholds like the ones often used in the literature, for example a minimum length of 12 nt (12, 6, 4, 3, 3, 3 repeats, respectively) (used by EDWARDS *et al.* 1998; GUO *et al.* 2009; JURKA and PETHIYAGODA 1995; MORGANTE *et al.* 2002; SUBRAMANIAN *et al.* 2003; TOTH *et al.* 2000), the proportion of missing hits diminishes below 10 % for the sequences used for the tests. With this higher threshold it was feasible to run the IrSa program on the human chromosome 22, and the comparison of this dataset with TRF output showed that the proportion of numbers and coverage of hits missed by TRF reduces dramatically at higher minimum length thresholds, in this case down to 0.73% for the joint TRF236+trf277 dataset (**table 3.5**). This indicates that TRF is more efficient in detecting longer imperfect hits, although at the expense of missing a proportion of shorter tandem repeats.

Table 3.4: Characteristics of the test sequences in comparison to the percent of microsatellite coverage missed by TRF.

Test sequence	Organism	Sequence length	%CG	% coverage	% imperfection	% coverage of missed hits
NC_003997.fa	*Bacillus antracis* str. Ames	5227293	35.38	7.7%	99.38	44.42
yeast1.fa	*Saccharomyces cerevisiae* (Chr I)	230208	39.75	5.01	57.96	38.87
danio.fa	*Danio rerio* (Chr 1: 1-13442)	13442	31.47	10.28	58.55	19.57
plas1.fa	*Plasmodium falciparum* (Chr1	643292	20.55	21.98	42.91	14.03

Table 3.5: Proportion of missing perfect microsatellites in TRF search results for the human chromosome 22. The parameter sets are: TRF236=2 3 6 75 20 10 6, TRF277=2 7 7 80 10 10 6, filtered for minimum lengths of 12, 6, 4, 3, 3 and 3 repeats and a minimum of 60% matches. Redundant hits were also filtered out previous to the comparisons.

Parameters	Number of hits	coverage
IrSa empirical perfect repeats	28450	503288
TRF236	115861	2415415

TRF277	43639	1009391		
TRF236+TRF277	119590	2506961	Proportion of missing perfect hits	Percentage coverage of missing hits
IrSa – TRF236	1039	14846	3.65	2.95
IrSa – TRF277	345	5257	1.21	1.04
IrSa – TRF236+TRF277	207	2999	0.73	0.60

3.3.2 SciRoKo

SciRoKo is a program specialized to find tandem repeats with motifs from 1 to 6 nt in length. This motif length can not be modified, for example, to look only for dinucleotides. However, motif types of interest can be analyzed individually afterwards in the SSR statistic module of the SciRoKo GUI version. Moreover, having to run complete searches for all motifs every time is not disadvantageous because the execution speed of the program is the highest among all programs tested in Chapter II. Execution time will usually depend on the query sequence length and the search parameters used, as well as on the overall content of microsatellites in the sequence. During the tests the mean execution time on human genome chromosomes ranged from 1.5 to 5.6 Mb per second, correlating positively with the number and coverage of microsatellites reported (**figure 3.7**).

The first public version of SciRoKo, version 3.1, could not run directly through whole chromosomes. The chromosomes had to be digested into sequence chunks with a helper tool prior to the microsatellite search. This problem has been overcome in version 3.3, which is the version used here. SciRoKo could process any of the chromosomes published so far, except for the largest chromosomes from the opossum genome (chromosomes 1 and 2, with lengths of 733 and 528 Mb, respectively.

SciRoKo can report perfect as well as imperfect repeats, and it allows using different minimum length threshold measurements for the search: an absolute minimum length in nt with the pl and mismatched modes, and a motif-relative minimum lenght in number of repeats in the pr and misa modes. In mismatched modes, the minimum length depends mainly on the score, but the initial length and thus the required initial score for the microsatellite hits is set by the –seedr (number of repeats required in the search seed). and –seedl (number of nucleotides required in the search seed) parameters. The lower the values for these parameters, the shorter, more imperfect and more abundant hits were obtained in the test runs. The minimum number of repeats can also be given priority over the length in the mismatched modes by using the –seedr parameter. However, it is not possible to assign

a –seedr value individually for each motif, and therefore using it in combination with the –seedl value is still the best option to avoid excessively short mono- and dinucleotides.

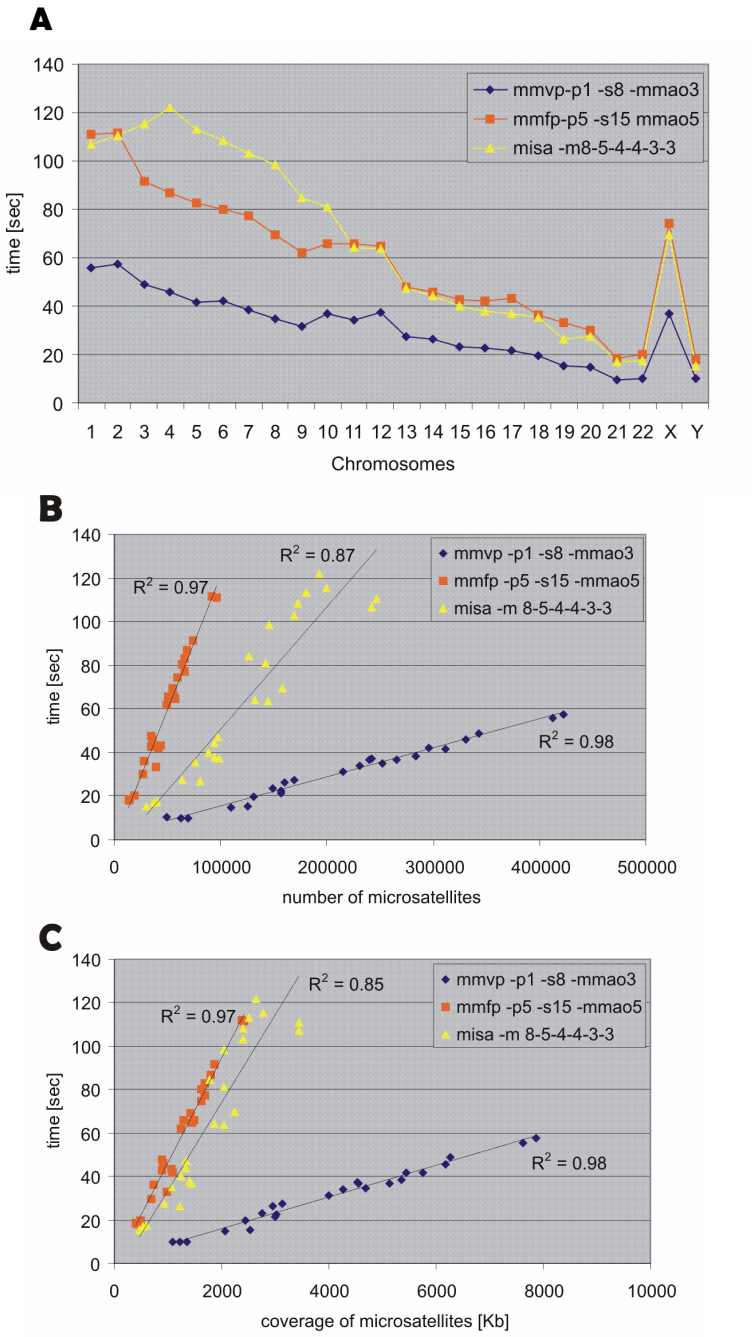

Figure 3.7: Execution times of program SciRoKo with three representative parameter sets on the human chromosomes. Graph A shows the difference among three modes with different parameters. The parameters that are not mentioned were used with default values: -seedl 8, -seedr 3. Average execution times for the parameter sets shown are 4.24 Mb/sec for the mmvp mode, 2.21 Mb/sec for the mmfp mode, and 2.16 Mb/sec for the misa mode. These are not mean values for the search modes *per-se*, as the execution time will depend strongly on the specific parameter values, which influence directly the number

and coverage of microsatellites detected. Overall the number and coverage of microsatellites detected correlate positively with the execution time of the program, as shown in graphs B and C.

There were no redundant hits in the output of SciRoKo, but the process by which the redundancy is reduced is not explained in the program's documentation. Moreover, the ScoRoKo algorithm considers that every nucleotide in the query sequence can correspond to only one microsatellite. In case of overlaps among two microsatellite hits, the overlapping nucleotides are considered to be part of the first registered hit (the 5'-most hit), and excluded from the second overlapping hit. Therefore, the second hit is either reported as a shorter hit, or otherwise excluded from the output if it becomes too short without the overlapping nucleotides. This became evident during the comparison among output datasets, using subtraction operations in the Galaxy webpage (obtaining "non-overlapping pieces of intervals"). The comparisons were performed first with a minimum length of 3 repeats for all motifs (**table 3.6**), and subsequently increasing the minimum length of mononucleotides to 5 repeats (**table 3.7**), because the bulk of detected tandem repeats at a minimum of 3 repeats corresponded to mononucleotides with 3 and 4 repeats (63 to 84% of microsatellites representing 34 to 71% in coverage, depending on the sequence). The numbers of missing hits for SciRoKo –pr3 and SciRoKo –misa 5-3-3-3-3-3 in comparison to the empirically obtained datasets with minimum lengths of 5,3,3,3,3,3 repeats for mono- to hexanucleotides, respectively, are shown in tables **3.6** and **3.7** for four of the test sequences. A higher number of hits was missing when running SciRoKo with a minimum microsatellite length of 3 repeats (-pr3 **table 3.6**), because in this case more shorter hits are reported, and the probability of these being adjacent to other hits, or part of these hits, is also higher.

The proportion of microsatellites misdetected or not detected by SciRoKo decreases when higher minimum length thresholds are used, and when searches are performed in mismatched modes. However, the number of microsatellties missing from the searches will strongly depend on the proportion of adjacent microsatellites within the sequences analyzed. If this proportion varies significantly among species, then using SciRoKo for inter-genomic comparisons of microsatellite abundance can produce misleading results.

Table 3.6: Microsatellite hits missed by SciRoKo in pr mode due to overlaps with shorter hits.

		danio.fa	NC_003997.fa	plas1.fa	yeast1.fa
Empyrical min 5-3-3-3-3-3	number *	224	56993	17391	2183
	coverage	2361	279327	147819	12237
SciRoKo pr 3	Number	779	345500	50712	13844
	coverage	3435	921863	213089	37866
number of hits with 3 repeats not reported		13	2075	2538	73
number of hits reported shorter		45	1103	1149	90
Total coverage of missed repeats or parts of repeats		123	12728	8902	629
%coverage of missed repeats		5.21	4.56	6.02	5.14

* the numbers of microsatellites in the empirical datasets were merged: adjacent microsatellites were joined into one hit. Therefore, these numbers may not be compared directly, but the coverage value needs to be taken into account.

Table 3.7: Microsatellite hits or parts of hits missed by SciRoKo in misa mode due to overlaps with shorter hits.

		danio.fa	NC_003997.fa	plas1.fa	yeast1.fa
Empyrical min5-3-3-3-3-3	number *	224	56993	17391	2183
	coverage	2361	279327	147819	12237
SciRoKo MISA 5-3-3-3-3-3	number	246	57244	18648	2200
	coverage	2286	276144	142676	12057
number of hits with 3 repeats not reported		10	611	742	30
number of hits reported shorter		22	240	1149	15
Total coverage of missed repeats or parts of repeats		75	3194	5174	192
%coverage of missed repeats		3.18	1.14	3.50	1.57

* the numbers of microsatellites in the empirical datasets were merged: adjacent microsatellites were joined into one hit. Therefore, there numbers may not be compared directly, but the coverage value needs to be taken into account.

The two mismatched search modes from SciRoKo differ in the way the score is calculated, as explained in the methods section. The microsatellite number and coverage distributions obtained by varying score and seedl while keeping the parameters -p, -seedr and -mmao constant with the values 1, 3, and 3, respectively, are shown in **figure 3.8.** For the same range of parameter settings the mmfp generates slightly less hits in the output, but the coverage is much higher than for the mmvp mode, which shows that the hits are much longer in the mmfp-derived results. These differences come as a result of the presence of adjacent microsatellites in the query sequences; microsatellites with different motifs that are very close together. In mmfp mode, SciRoKo will extend through microsatellites with different motifs as long as the periodicity is maintained in a very similar way to TRF because, in both cases the score is not influenced by the motif length. On the other hand, in mmvp mode, the motif length is multiplied by the mismatch penalty for the score calculation, and hence the penalties have a different effect on the score depending on the motif length. Therefore, adjacent microsatellites with different motif lengths are usually reported individually in mmvp mode, unless the nucleotide composition of the corresponding motifs is similar, in which case two different scenarios can occur, as shown in **table 3.8**. If the 5'-most microsatellite has a shorter motif than its 3' adjacent microsatellite, the two microsatellites will be merged in a single hit and reported with the shorter motif. However, if the 5'-most microsatellite has the longest motif, the adjacent microsatellites will be most likely reported in separate hits with their respective motifs. This occurs because the reduction in score due to mismatches will be directly proportional to the motif length, rendering longer motifs less tolerant to imperfections than shorter ones.

Table 3.8: Unequal merging of adjacent microsatellites in SciRoKo mmvp reports**.** When the 5'-most motif is shorter, the microsatellties are merged and reported under the first detected motif. If the 5'-most motif is longer than the subcequent ones, are reported in independent hits.

305	340	AAAG	AAAGAAAGAAAGAAAGAAAAGAAAGAAAGAAAGAAA
341	371	AG	AGAGAGAGAGGAGGAGGAGGAGGAGAGAGAG
657	683	GTG	GTGTGTGTTGGTTGGTTGGTTGGTGGTG
685	736	CT	CTCTCTCTCCTCCTCCTCCTCCTCTCTCTTCTCTCTCTTCTTCTTCTTCTTC

From a biological point of view, it would not be realistic to attempt separating all adjacent microsatellites into single hits. Based on the structure of microsatellite clusters, it is likely that adjacent microsatellites with different motifs are generated due to substitutions

or indels within long imperfect microsatellites, in which case it would be reasonable to report the whole cluster with the original motif, which will usually be the most abundant one. In contrast, it may be of interest to quantify compound and/or adjacent microsatellites, in which case it would be important to make sure that the program used for the searches can quantify all possible microsatellite motif combinations with the same probability. The authors of SciroKo published the first in-depth analysis of microsatellite clustering in eight eukaryotic genomes by using SciRoKo with parameters mmfp –p5 –s15 –seedl8 –seedr3 –mmao5 (KOFLER *et al.* 2008). Although the mmfp mode tends to join adjacent microsatellites regardless of the motif, this hit merging was probably reduced by the high mismatch penalty value (-p 5) used.

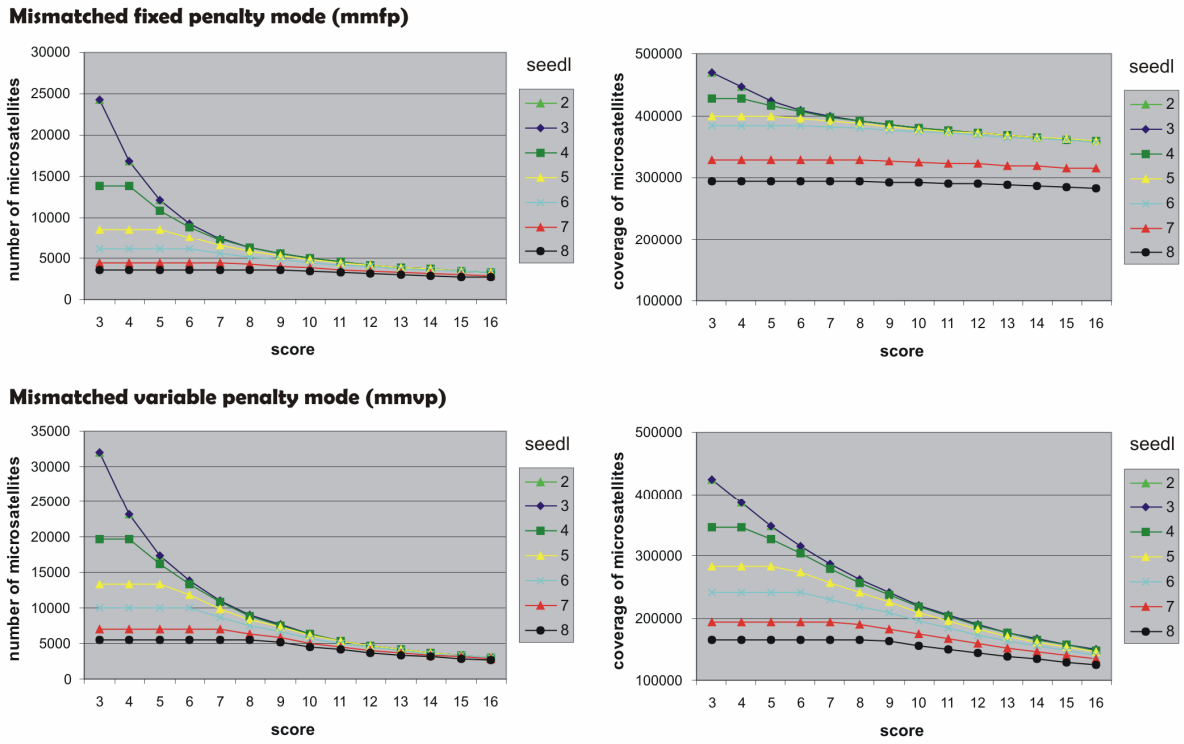

Figure 3.8: Comparison of microsatellite number and coverage distributions between the mmfp and mmvp search modes from SciRoKo. The additional search parameters not mentioned in the legends are: -p 1, -seedl 3, -mmao 3. Both modes produce very similar number of microsatellite distributions, but the coverage distributions show much higher values for the mmfp mode. SciRoKo in mmfp mode tends to extend through imperfections regardless of a change of motif, therefore reporting very long hits representing microsatellite clusters. In mmvp different motifs are, in most cases, reported in different hits, producing therefore a higher number of hits, but less coverage because the sequence segments between clustered microsatellites are not included.

The 'maximum mismatches at once parameter', mmao, does not refer to an absolute amount of indels allowed in the whole microsatellite; it denotes the absolute number of substitutions and indels allowed between segments of the same microsatellite; if the mmao value is reached before additional repetitions are detected, the search is stopped and the hit is reported to the output. The mmao parameter could therefore be interpreted as the maximum fragmentation allowed within microsatellite hits. Usually, this amount of imperfection is controlled with the score, and tandem repeat finding programs continue checking nucleotide positions until the score falls below a threshold. Restricting this parameter aids the efficiency of the program, because it reduces dramatically the amount of nucleotides to process per microsatellite. However, it is not the ideal solution, because very large imperfect clusters of microsatellites like the ones shown in figures **3.1** to **3.3** will be broken down into several pieces based on an arbitrary value instead of depending on the length and composition of the microsatellite. Moreover, this is the critical parameter determining the efficiency of SciRoKo. With the default mmao value of 3, SciRoKo can process whole ~250 Mb chromosomes in a matter of seconds. However, the time required for the search increases exponentially when increasing this parameter (**figure 3.9**), and the maximum value that could be used during the test searches was 16. With an mmao VALUE of 16, the search on a sequence as small as danio.fa (13442 nt) took half an hour to complete. Therefore, using values higher than 6 for the mmao parameter would render the program highly inefficient for searches through large eukaryotic genomes.

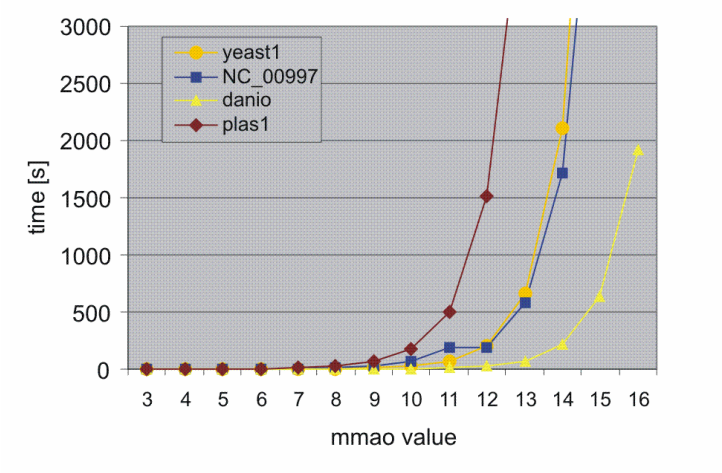

Figure 3.9: Exponential increase in execution time of the program SciRoKo with respect of the mmao value. The parameters used in these searches were: -mode mmao –p 1 –s 8 –seedl 6 –mmao x.

3.3.3 TRF *vs* SciRoKo

An immediately evident difference among the programs TRF and SciRoKo are the execution times. SciRoKo was the fastest algorithm among the programs tested in Chapter II, and a comparison in execution times with TRF can be observed in **figure 3.10**.

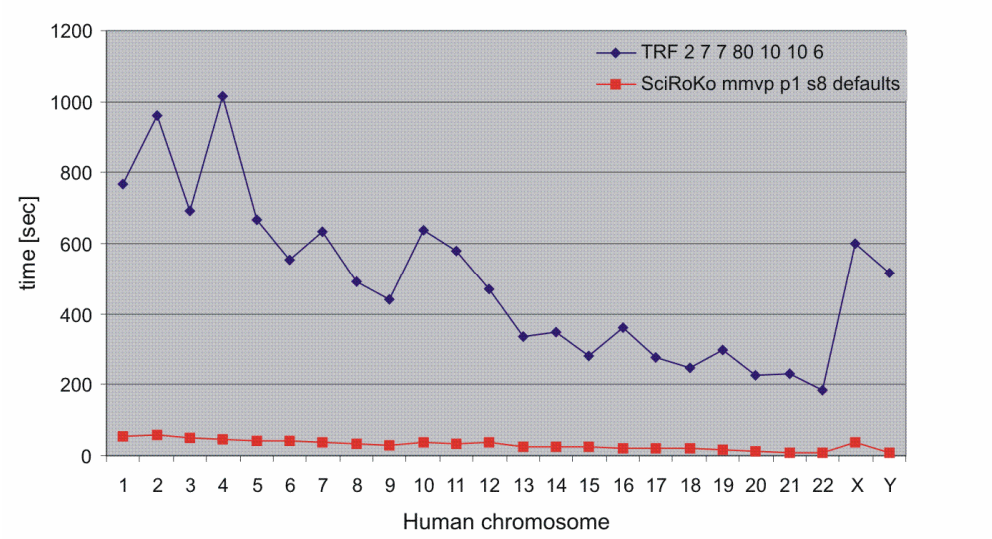

Figure 3.10: Comparison of TRF and SciRoKo execution times on human chromosomes. The parameters used are the most stringent ones for TRF while SciRoKo was used in mismatch-variable-penalty (mmvp) mode with mismatch penalty (-p) reduced to 1. Clearly, SciRoKo is much more efficient in terms of execution time.

TRF and SciRoKo are based on essentially different search algorithms. SciRoKo is specific for the search of microsatellites, while TRF can find tandem repeats with motifs 1 to 2000 nt in length. For this reason, in TRF the scores are evaluated with respect to the total length of the tandem repeat, which is very important for longer motif sizes. Due to this relative scoring system the extension of tandem repeat sequences through imperfections is highly efficient with TRF. However, the verification process, which is based on sequence alignments with a perfect tandem repeat version of the consensus motif, is highly demanding, especially for long imperfect microsatellites. This may be the reason why, the more relaxed parameters are used, the higher is the proportion of small tandem repeat hits missed by TRF

Unlike TRF, the program SciRoKo does not include sequence alignment processes in its algorithm, which is likely to be one of the main reasons for its high search efficiency. The SciRoKo Manual (KOFLER *et al.* 2007a) mentions comparisons with a 'virtual perfect microsatellite (vpm)' aiding the scoring process. However, this virtual microsatellite is used

directly during the search as a mean to 'predict' what the next nucleotide would have to be to extend the microsatellite further, recognizing in this way if an insertion or a substitution has occurred. This information is only used to fine-tune subsequent predictions (i.e. to virtually slide the vpm towards the 5' or 3' end; (see figure 1 in KOFLER *et al.* 2007a)), but not to make a difference among mismatches and indels in the scoring process. The content of interruptions within microsatellite hits in the SciRoKo output is primarily dependent on the score, and also on the parameter mmao, which refers to the maximum number of mismatches (both substitutions and indels) which are allowed to occur before breaking the microsatellite into two hits. Although the mmao value does not apply to whole tandem repeat hits, but only to the fragments within it, it is still designating an absolute gap length which will affect differently tandem repeats with different motif lengths; shorter motifs will be allowed longer gaps with respect to the repeat number than longer motifs, producing either too imperfect mono-, di- and trinucleotides, or otherwise an underrepresentation of imperfect tandem repeats with longer motifs. For this reason, requiring a fixed number of differences within a microsatellite hit rather than a percentage is regarded as unsatisfactory (see DE RIDDER *et al.* 2006).

A comparison of the main execution and output characteristics among TRF and SciRoKo is presented in **table 3.9**. The most distinctive characteristic of the program TRF was its capacity to extend microsatellite hits through imperfections with great efficiency, therefore reporting longer hits than SciRoKo for equivalent microsatellite definitions. This became evident when comparing the imperfect microsatellite hits reported by TRF and SciRoKo. An example figure of the comparison of microsatellite hits among TRF and SciRoKo datasets produced with various combinations of parameter settings for similar microsatellite definitions can be seen in **figure 3.11**. The program SciRoKo, on the other hand, showed a better capacity than TRF to identify small perfect hits. Therefore I decided that using both programs in a complementary way can produce more complete microsatellite datasets.

The SciRoKo searches in mmvp mode with parameters –p 1 –s 8 –seedl6 –seedr 3 –mmao 9 produced the best results in all sequences tested. Reducing the mmao parameter allows for faster searches while fragmenting very long imperfect microsatellites into several pieces. However, this does not produce a loss in detection power. Therefore, a mmao value of 6 was chosen for searches in larger chromosomes. Longer clusters of microsatellites can then be identified through intersection with the results from the program TRF.

Table 3.9: Comparison of execution and output characteristics between TRF and SciRoKo.

Characteristic	TRF	SciRoKo
Average speed *	~0.026 to 0.26 Mb/sec	~2.16 to 4.24 Mb/sec
Redundancy	Can produce up to 6 redundant detections for 2 to 12% of hits	No redundancy reported.
Completeness of results	Can miss more than 50% of hits.	Almost exhaustive at short sequence lengths (<5 Mb).
Output	Complete set of microsatellite characteristics, very easy to filter.	Limited, not easy to filter or manipulate.
Capacity to extend through imperfections and interruptions	Excellent	Very good; in mmfp mode this capacity is similar to TRF, extending through whole microsatellite clusters.
Capacity to distinguish between adjacent microsatellites with different motifs	Poor, it does not improve much by increasing stringency.	Poor with mmfp mode. Good with mmvp mode, because the score calculation depends on the motif length.
Substitutions vs indels	Can be assigned different penalties (second and third parameters). The effect of these will be relative to the total length of the hit.	Substitutions and indels are not distinguished in the scoring system. They are both referred to as mismatches and dealt with through the mismatch penalty (-p) parameter. The amount of mismatches allowed in a row is restricted by a fixed value, the mmao parameter, and the maximum value which could be assigned to this value before crashing the program was 15.
Motif-specific score calculation	No	Yes with the mmvp mode.
Motif standarization	None	Partial and complete
Statistical analysis	None	Includes a statistical module

* The speed will depend on the parameters used, on the query sequence length, and on the abundance and complexity of microsatellites in the query sequence. The values presented here are based on the rest sequences and on the human chromosomes (Hg18)

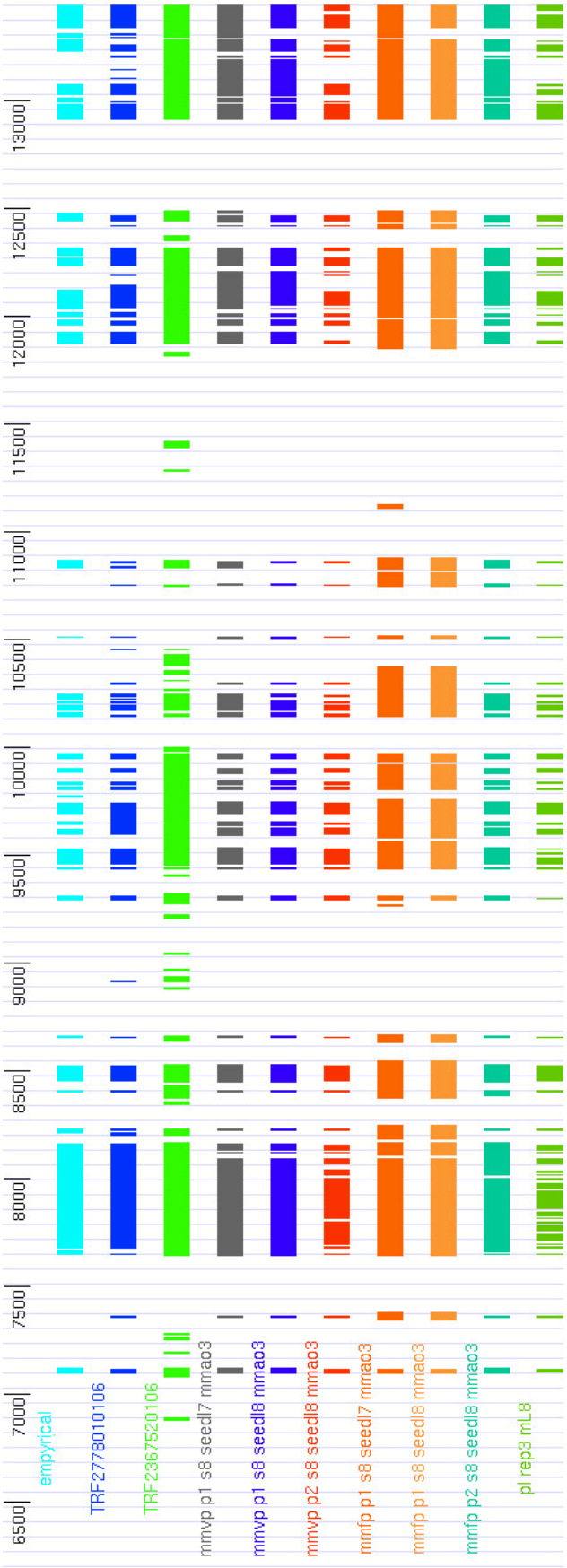

Figure 3.11: Comparison of hits obtained with various TRF and SciRoKo search parameters on the sequence danio.fa. The first row (track) shows the empirical microsatellite dataset (from IrSa), followed by two TRF datasets, and seven SciRoKo datasets. The text in each row shows the parameters used for each search. All datasets except the last one span imperfect repeats. The last dataset, obtained with the pl mode of SciRoKo with a minimum length of 8 nt, shows only the perfect tandem repeat segments, and therefore the hits are shorter and more abundant. The two TRF datasets show a contrast of the most stringent and the most relaxed parameters that can be set in TRF, respectively (too short hits, less than 7, 5, 3, 3, 3, 3, for mono- to hexanucleotides, were filtered out from the datasets). TRF extended hits through more imperfections than the ones contained in the empirical dataset, showing therefore fewer and longer hits. The second TRF track finds additional hits to the ones in the empirical dataset; these were too imperfect to spot by eye. SciRoKo datasets tend to contain more hits than TRF because the long imperfect microsatellites tend to be reported as several hits instead of one (especially with the low mmao value used to obtain these datasets.

TRF and SciRoKo are heuristic tandem repeat finding programs. Therefore, they don't attempt to exhaustively identify all tandem repeats within a sequence. Instead, the generated datasets are meant to contain a representative subset of the tandem repeats which satisfy the search parameter conditions. Therefore, it is expectable to observe differences between the output datasets and the 'real' microsatellite content of a sequence, as I showed here in the comparisons of TRF and SciRoKo datasets with empirical microsatellite datasets.

The programs TRF and SciRoKo present several very useful features for approximate microsatellite identification, but they also have some drawbacks. These two programs were already selected from a pool of available tandem repeat finders tested in Chapter II, not necessarily because they offered the "best" results, but mainly because they offered the best usability, appropriate flexibility in search parameters, and relatively complete documentation. Any program may find a useful application as long as it offers enough information on its useful features as well as its drawbacks. Otherwise, this information needs to be "rediscovered" by the user to be able to make sense of the results.

Once known, the drawbacks from TRF and SciRoKo can either be overcome, for example, with additional filters for redundancy and false positive hits, or otherwise taken into account in the result interpretation. The proportion of missing hits in the output can be minimized by joining output datasets obtained with different parameter sets and/or programs. In any case, however, when presenting the results of a microsatellite search, detailed information of the methodology used should be presented so that the results can be compared or reproduced in other studies.

3.4 Conclusions

The programs TRF and SciRoKo are heuristic algorithms and therefore do not search exhaustively for all tandem repeats in a query sequence. Both programs miss certain proportions of hits depending on the query sequence and on the program parameters. During the tests performed here, the missing hits amounted up to 15% for SciRoKo in perfect search mode, and up to 70% for TRF with the most restrictive parameters (2 7 7 80 10 10 6).

The majority of hits missed by TRF are short tandem repeats with short motifs, and therefore the detections improve dramatically when higher minimum length thresholds are used for the searches (e.g. 12 or 15 nt). However, the search runs should still be performed with the short minscore value of 10, to subsequently filter the datasets with complementary scripts presented here.

TRF is very efficient for detecting long highly imperfect microsatellites or microsatellite clusters, for which the parameters 2 3 6 75 20 10 6, as given into the program, gave the best results. To account for missing hits, these results are best complemented by joining them with output from a more restrictive TRF search, with parameters 2 7 7 80 10 10 6, or from a SciRoko run in mismatch variable penalty mode with a mismatch penalty of 1 (-p), score of 8, seed length of 8, mmao of 6, and the seed repeats with the default parameter 3.

TRF results need to be filtered to eliminate redundancy and excessively small or imperfect repeats. Since the redundancy needs to be filtered anyway, it is best to filter minimum lengths and imperfection after the TRF run, while leaving the corresponding parameters at very relaxed parameters for the TRF search. This improves the overall capacity of the program to detect both small and long microsatellites.

The program SciRoKo will report microsatellite clusters as several shorter hits. These results would either need to be post-processed to group hits by proximity in the sequence, or otherwise complemented with the results from TRF.

3.5 References

AMES, D., N. MURPHY, T. HELENTJARIS, N. SUN and V. CHANDLER, 2008 Comparative analyses of human single- and multilocus tandem repeats. Genetics **179:** 1693-1704.

ARCHAK, S., E. MEDURI, P. S. KUMAR and J. NAGARAJU, 2007 InSatDb: a microsatellite database of fully sequenced insect genomes. Nucleic Acids Res **35:** D36-39.

BENSON, G., 1999 Tandem repeats finder: a program to analyze DNA sequences. Nucleic Acids Res **27:** 573-580.

BOYER, J. C., N. A. YAMADA, C. N. ROQUES, S. B. HATCH, K. RIESS *et al.*, 2002 Sequence dependent instability of mononucleotide microsatellites in cultured mismatch repair proficient and deficient mammalian cells. Hum Mol Genet **11:** 707-713.

BRANDSTROM, M., and H. ELLEGREN, 2008 Genome-wide analysis of microsatellite polymorphism circumventing the ascertainment bias. Genome Res.

CASACUBERTA, E., P. PUIGDOMENECH and A. MONFORT, 2000 Distribution of microsatellites in relation to coding sequences within the *Arabidopsis thaliana* genome. Plant Sci **157:** 97-104.

CASTELO, A. T., W. MARTINS and G. R. GAO, 2002 TROLL--tandem repeat occurrence locator. Bioinformatics **18:** 634-636.

CHAMBERS, G. K., and E. S. MACAVOY, 2000 Microsatellites: consensus and controversy. Comp Biochem Physiol B Biochem Mol Biol **126:** 455-476.

COLLINS, J. R., R. M. STEPHENS, B. GOLD, B. LONG, M. DEAN *et al.*, 2003 An exhaustive DNA micro-satellite map of the human genome using high performance computing. Genomics **82:** 10-19.

CROLLIUS, H. R., O. JAILLON, C. DASILVA, C. OZOUF-COSTAZ, C. FIZAMES *et al.*, 2000 Characterization and repeat analysis of the compact genome of the freshwater pufferfish *Tetraodon nigroviridis*. Genome Res. **10:** 939-949.

DE RIDDER, C., D. KOURIE and B. WATSON, 2006 FireμSat: An algorithm to detect microsatellites in DNA, pp. in *Proceedings of the Prague Stringology Conference 2006*, Prague.

DOXIADIS, G. G., N. DE GROOT, F. H. CLAAS, DOXIADIS, II, J. J. VAN ROOD *et al.*, 2007 A highly divergent microsatellite facilitating fast and accurate DRB haplotyping in humans and rhesus macaques. Proc Natl Acad Sci U S A **104:** 8907-8912.

EDWARDS, Y. J., G. ELGAR, M. S. CLARK and M. J. BISHOP, 1998 The identification and characterization of microsatellites in the compact genome of the Japanese pufferfish, *Fugu rubripes*: perspectives in functional and comparative genomic analyses. J Mol Biol **278:** 843-854.

FUJIMORI, S., T. WASHIO, K. HIGO, Y. OHTOMO, K. MURAKAMI *et al.*, 2003 A novel feature of microsatellites in plants: a distribution gradient along the direction of transcription. FEBS Lett **554:** 17-22.

GIARDINE, B., C. RIEMER, R. C. HARDISON, R. BURHANS, L. ELNITSKI *et al.*, 2005 Galaxy: A platform for interactive large-scale genome analysis. Genome Res. **15:** 1451-1455.

GUO, W., C. CAI, C. WANG, Z. HAN, X. SONG *et al.*, 2007 A microsatellite-based, gene-rich linkage map reveals genome structure, function, and evolution in *Gossypium*. Genetics **176:** 527-541.

GUO, W. J., J. LING and P. LI, 2009 Consensus features of microsatellite distribution: Microsatellite contents are universally correlated with recombination rates and are preferentially depressed by centromeres in multicellular eukaryotic genomes. Genomics.

HALL, T. A., 1999 BioEdit: a user-friendly biological sequence alignment editor and analysis program for Windows 95/98/NT. Nucl. Acids. Symp. Ser. **41:** 95-98.

HARR, B., B. ZANGERL and C. SCHLÖTTERER, 2000 Removal of microsatellite interruptions by DNA replication slippage: phylogenetic evidence from *Drosophila*. Mol Biol Evol **17:** 1001-1009.

JURKA, J., and C. PETHIYAGODA, 1995 Simple repetitive DNA sequences from primates: compilation and analysis. J Mol Evol **40:** 120-126.

KARACA, M., M. BILGEN, A. N. ONUS, A. G. INCE and S. Y. ELMASULU, 2005 Exact tandem repeats analyzer (E-TRA): a new program for DNA sequence mining. J Genet **84:** 49-54.

KAROLCHIK, D., R. BAERTSCH, M. DIEKHANS, T. S. FUREY, A. HINRICHS *et al.*, 2003 The UCSC Genome Browser Database. Nucl. Acids Res. **31:** 51-54.

KATTI, M. V., P. K. RANJEKAR and V. S. GUPTA, 2001 Differential distribution of simple sequence repeats in eukaryotic genome sequences. Mol Biol Evol **18:** 1161-1167.

KOFLER, R., C. SCHLÖTTERER and T. LELLEY, 2007a The SciRoKo 3.1 Manual, pp., Vienna, Austria.

KOFLER, R., C. SCHLÖTTERER and T. LELLEY, 2007b SciRoKo: A new tool for whole genome microsatellite search and investigation. Bioinformatics **23:** 1683-1685.

KOFLER, R., C. SCHLÖTTERER, E. LUSCHUTZKY and T. LELLEY, 2008 Survey of microsatellite clustering in eight fully sequenced species sheds light on the origin of compound microsatellites. BMC Genomics **9:** 612.

KRUGLYAK, S., R. DURRETT, M. D. SCHUG and C. F. AQUADRO, 2000 Distribution and abundance of microsatellites in the yeast genome can be explained by a balance between slippage events and point mutations. Molecular Biology and Evolution **17:** 1210-1219.

LECLERCQ, S., E. RIVALS and P. JARNE, 2007 Detecting microsatellites within genomes: significant variation among algorithms. BMC Bioinformatics **8:** 125.

LEVINSON, G., and G. A. GUTMAN, 1987 High frequencies of short frameshifts in poly-CA/TG tandem repeats borne by bacteriophage M13 in *Escherichia coli* K-12. Nucleic Acids Res **15:** 5323-5338.

MALPERTUY, A., B. DUJON and G. F. RICHARD, 2003 Analysis of microsatellites in 13 hemiascomycetous yeast species: mechanisms involved in genome dynamics. J Mol Evol **56:** 730-741.

MERKEL, A., and N. GEMMELL, 2008 Detecting short tandem repeats from genome data: opening the software black box. Brief Bioinform **9:** 355-366.

MORGANTE, M., M. HANAFEY and W. POWELL, 2002 Microsatellites are preferentially associated with nonrepetitive DNA in plant genomes. Nat Genet **30:** 194-200.

SHARMA, P. C., A. GROVER and G. KAHL, 2007 Mining microsatellites in eukaryotic genomes. Trends Biotechnol.

SUBRAMANIAN, S., R. K. MISHRA and L. SINGH, 2003 Genome-wide analysis of microsatellite repeats in humans: their abundance and density in specific genomic regions. Genome Biol **4:** R13.

SYMONDS, V. V., and A. M. LLOYD, 2003 An analysis of microsatellite loci in *Arabidopsis thaliana*: Mutational dynamics and application. Genetics **165:** 1475-1488.

TAUTZ, D., M. TRICK and G. A. DOVER, 1986 Cryptic simplicity in DNA is a major source of genetic variation. Nature **322:** 652-656.

TOTH, G., Z. GASPARI and J. JURKA, 2000 Microsatellites in different eukaryotic genomes: survey and analysis. Genome Res. **10:** 967-981.

UDAR, N., S. FARZAD, L. Q. TAI, J. O. BAY and R. A. GATTI, 1999 NS22: a highly polymorphic complex microsatellite marker within the ATM gene. Am J Med Genet **82:** 287-289.

VAN OPPEN, M. J., C. RICO, G. F. TURNER and G. M. HEWITT, 2000 Extensive homoplasy, nonstepwise mutations, and shared ancestral polymorphism at a complex microsatellite locus in Lake Malawi cichlids. Mol Biol Evol **17:** 489-498.

VAN TREUREN, R., H. KUITTINEN, K. KARKKAINEN, E. BAENA-GONZALEZ and O. SAVOLAINEN, 1997 Evolution of microsatellites in *Arabis petraea* and *Arabis lyrata*, outcrossing relatives of *Arabidopsis thaliana*. Mol Biol Evol **14:** 220-229.

WARBURTON, P. E., D. HASSON, F. GUILLEM, C. LESCALE, X. JIN *et al.*, 2008 Analysis of the largest tandemly repeated DNA families in the human genome. BMC Genomics **9:** 533.

WOOD, S., and M. SCHERTZER, 1992 A polymorphic complex dinucleotide repeat at the telomeric D8S7 locus. Hum Hered **42:** 149-152.

YERAMIAN, E., and H. BUC, 1999 Tandem repeats in complete bacterial genome sequences: sequence and structural analyses for comparative studies. Res Microbiol **150:** 745-754.

CHAPTER IV: The Minimum Length Threshold for Microsatellite Identification

Abstract

In Chapter II I showed that the minimum microsatellite length is a key parameter to define microsatellite searches. Interestingly, in the biological definition of microsatellites, there is no real agreement regarding the minimum length necessary for tandem repeats to become prone to microsatellite-fashioned mutations. Nevertheless, the classical definition of microsatellites implies that these are overrepresented within genomic sequences due to their frequent expansions. Thus, the minimum length threshold for a tandem repeat to be considered a micrtosatellite will be the minimum number of repetitions at which an overrepresentation is first detected. To find out what this threshold for overrepresentation is, and if this threshold is the same among different genomes, I applied two models, a probabilistic model published by de Wachter in 1981, and a second order Markov model, to calculate the expectations of tandem repeat abundance within 24 eukaryotic, 8 prokaryotic, and 5 archaeal genomes. By comparing the modelled expectations for mono-, di-, and trinucleotide microsatellites with the corresponding microsatellite frequencies observed for each genome, I show that the minimum length threshold for a microsatellite to become overrepresented can vary significantly depending on the microsatellite motif type and on the species the genome belongs to. These differences are probably due to divergence in metabolism and replication dynamics across different taxa, which would question the appropriateness and biological relevance of the standard practice of applying the same minimum length threshold for microsatellite searches for different species. My results suggest that the best practice when searching for microsatellites in genomic sequences is to develop a null expectation for microsatellite content, as presented here, to determine minimum microsatellite length thresholds in every newly published genome or genome build before performing microsatellite searches. Such an approach could be built into search pipelines and would lead to a more robust approach to determine if particular short tandem repeats are likely to describe microsatellite-like behaviour or not.

4.1 Introduction

The minimum length or minimum number of repeats for a short tandem repeat to be considered as a microsatellite is a critical parameter because of its influence on the sensitivity and runtime of microsatellite search algorithms: if the threshold value is too high, valid microsatellite candidates could be missed, while if this threshold is too small, the possibility of reporting false positives is increased. Further, the smaller the minimum length of tandem repeats chosen, the more data the program has to process, which, in the case of large sequences, can cause search programs to run out of memory and crash (DOMANIC and PREPARATA 2007). To overcome this issue, most researchers assign relatively high values for the minimum microsatellite length: the preferred value is 12 nucleotides (see EDWARDS *et al.* 1998; GUO *et al.* 2009; JURKA and PETHIYAGODA 1995; MORGANTE *et al.* 2002; SUBRAMANIAN *et al.* 2003; TOTH *et al.* 2000). This is not an ideal solution, but it is sufficient when the objective of the search is to find microsatellites to develop as molecular markers. A number of studies use lower thresholds for the minimum microsatellite length, for example a minimum of 2 repeats (COX and MIRKIN 1997; FIELD and WILLS 1998), 3 repeats (FUJIMORI *et al.* 2003; RAJENDRAKUMAR *et al.* 2007), or 5 repeats (BACHTROG *et al.* 1999; COENYE and VANDAMME 2005; LIM *et al.* 2004). Using minimum length thresholds in number of repeats makes this threshold relative to the motif length, unlike using the length in nucleotides, which is an absolute measure. A few authors even assign different thresholds in number of repeats for each motif analyzed. For example, Prasad *et al.* (2005) set the threshold at 15 repeats for mononucleotides, and 5 repeats for 2 to 6 nt motifs, and Webster *et al.*. (2002) set the thresholds at at 9, 5, 3, 3, and 2 repeats for mono-, di-, tri-, tetra- and pentanucleotides respectively. This last way to define the minimum microsatellite length is the most precise one, but the threshold values used are still very different, hindering direct comparisons among different studies. In order to infer microsatellite evolution patterns from genomic comparisons of microsatellite abundance and distribution, the minimum length thresholds would need to be set uniformly.

Biologically speaking, the length of microsatellite alleles is among the most important factors influencing microsatellite mutation rates (ELLEGREN 2004). It is widely accepted that microsatellite mutation rates are usually positively correlated with the total length of the microsatellite (BROHEDE *et al.* 2004; PRIMMER *et al.* 1996; WIERDL *et al.* 1997), and that among microsatellite mutations, expansions are usually more common than contractions (VIGOUROUX *et al.* 2003; XU *et al.* 2000). In this sense, microsatellites would tend to expand,

this tendency becoming reinforced every time a new expansion occurs. There would, however, need to be a threshold minimum length or minimum number of repeats above which short tandem repeats become susceptible to expansions by replication slippage and other mutation modalities of microsatellites, which may involve recombination (RICHARD and PAQUES 2000) and/or the formation of stable secondary structures within the tandem repeat array (BACOLLA *et al.* 2008; COX and MIRKIN 1997). Replication slippage, for example, is believed to depend on mispairing of tandem repeats during DNA replication (LEVINSON and GUTMAN 1987), and therefore it may not occur if there are too few repeats. Messier *et al.* (1996) suggested this threshold to be above 8 repeat units because microsatellites shorter than that are usually not polymorphic in humans (ARMOUR *et al.* 1994; VALDES *et al.* 1993). However, Zhu *et al.* (2000) showed that, although at lower rates, mutations generating duplications of repeats can still occur at very low repeat numbers and even in the absence of repeats, albeit at lower rates than for longer repeats.

Early in microsatellite history Tautz *et al.* (1986) introduced the notion that microsatellites evolve from 'cryptically simple sequences'. These are sequences where several directly repeated short motifs occur in close proximity. They compared the occurrence of these cryptically simple sequences within DNA sequences in the EMBL database and within randomized sequences, and concluded that all microsatellite motifs except A/T and C/G mononucleotides occurred more often than expected by chance. This overrepresentation of tandem repeats is expected to come about due to the dynamic mutations of microsatellites by replication slippage, which generates an excess of long repeat arrays (TAUTZ *et al.* 1986; ZHU *et al.* 2000, Rose, 1998 #190).

The overrepresentation of microsatellites was further examined in several papers. Rose and Falush (1998), using a probabilistic formula, concluded that tandem repeats below a threshold length of 8 nt were not more common than expected by chance in yeast. On the other hand, Pupko and Gaur (1999) also examined the yeast genome with a slightly different probabilistic formula published by de Wachter (1981) and concluded that microsatellites of any length are observed more frequently than expected by chance, and that hence there is no minimum length threshold. Metzgar *et al.* (2000), also applying the de Wachter model, found differences in the level of overrepresentation of microsatellites between coding and non-coding regions from DNA databases of various eukaryotic organisms. They presented individual minimum threshold values for each motif type and genome, and these range from 3 to 16 nucleotides.

Dechering (1998) and Marx (1993) used zero-order Markov models to calculate expected numbers of mononucleotide repeats in prokaryotic and eukaryotic microorganisms, and found that AT-rich mononucleotide repeats become overrepresented above 10 repeats, while CG-rich mononucleotides are usually not overrepresented. Lai and Sun (2003) also used two models, one based on Markov processes and the other one on branching processes, to examine all human chromosomes, coming up with a threshold of 9 repeat units for mononucleotides and 4 repeat units for longer motifs.

The consensus view from the previous studies is that there appear to be differences in the minimum length threshold for microsatellite mutations among different species and among motif types (i.e. mono-, di- and trinucleotides). While this could also be due to differences in the models used, it is reasonable to expect that the number of microsatellites of different motif types and motifs with different nucleotide composition should change based on the relative nucleotide compositions of the DNA sequences. Higher order composition patterns like dinucleotide and trinucleotide relative abundances may also affect these expectations differently among different genomes (GENTLES and KARLIN 2001; KARLIN and BURGE 1995). Thus, a new question arises regarding the appropriateness of using the same minimum length threshold for defining microsatellites in different genomes.

In Chapter II I discussed the importance of using equivalent search definitions (equivalent program parameter values) to be able to perform valid comparisons among microsatellite search results. However, although this would equalize the definitions used, it has not yet been tested if this could produce biases in microsatellite content comparisons. Therefore, in this Chapter I present an analysis of inter- and intra- genomic variation of microsatellite overrepresentation thresholds using two different probabilistic models: the de Wachter model (DE WACHTER 1981) and a second order Markov model. Based on these models I perform comparisons among the observed frequencies of perfect mono-, di- and trinucleotide tandem repeats in 24 eukaryotic, 8 prokaryotic, and 5 archaeal genomes, with their respective expected frequencies.

4.2 Methodology

The genomes analyzed in this Chapter were chosen based on their quality and completeness. Only fully assembled genome sequences were analyzed in order to obtain an unbiased representation of microsatellites in all genomic regions.

4.2.1 Calculation of the expected number of microsatellites based on sequence-specific motif frequency

Sequence-specific motif frequencies for each sequence analyzed were obtained by scanning query sequences with Java programs written in collaboration with Lisha Naduvilezhath, an exchange student from the Wolfgang Goethe University in Germany (programs MononuclFreq.java, DinuclFreq.java, TrinuclFreq.java, MononuclMarkov.java, DinuclMarkov.java, TrinuclMarkov.java). The frequencies of all nucleotide permutations from one to three nucleotides were counted for all chromosomes of the genomes listed in **tables 4.1** and **4.2**. The download sources, genome builds, and accession numbers for the bacteria and archaea are specified in **tables S1** to **S3** in the statement of sources. The mono-, di-, and trinucleotide frequencies were used to calculate the expected frequency of occurrence of tandem repetitions for each of the possible microsatellite motifs from one to three nucleotide long, consisting of two to ten repetitions (up to 20 and 15 repetitions for A/T and C/G mononucleotides, respectively), using two distinct models: a second order Markov model (KOSKI 2001), and a combinatory model proposed by de Wachter (1981).

Table 4.1: List of eukaryotic genomes for which the minimum microsatellite length threshold was analyzed

Eukaryotes	Scienfic name
Human	*Homo sapiens*
Chimp	*Pan troglodytes*
Rhesus monkey	*Macaca mulatta*
Cow	*Bos Taurus*
Horse	Equus caballus
Mouse	*Mus musculus*
Rat	*Ratus norvergicus*
Dog	*Canis familiares*
Opossum	*Monodelphis domestica*
Platypus	*Ornithorhynchus anatinus*
Chicken	*Gallus gallus*
Honeybee	*Apis mellifera*

Table 4.1: List of eukaryotic genomes for which the minimum microsatellite length threshold was analyzed (continued)

Eukaryotes	Scienfic name
Red flour beetle	*Tribolium castaneum*
Medaka	*Oryzias latipes*
Pufferfish	*Tetraodon nigroviridis*
Stickleback	*Gasterosteus aculeatus*
Zebrafish	*Danio rerio*
Fruitfly	*Drosophila melanogaster*
Mosquito	*Anopheles gambiae* str. PEST
Roundworm	*Caenorhabditis elegans*
Arabidopsis	*Arabidopsis thaliana*
Rice	*Oryza sativa* (japonica cultivar-group)

Table 4.2: List of archaeal and bacterial genomes for minimum microsatellite length threshold

Archaea
Hyperthermus butylicus DSM 5456
Methanocaldococcus jannaschii DSM 2661
Natronomonas pharaonis DSM 2160 and plasmids PL 131 and PL233
Pyrobaculum aerophilum str. IM2
*Methanosaeta thermophila*_PT
Bacteria
Neisseria meningitidis FAM18
Brucella melitensis biovar *Abortus* 2308
Bacillus anthracis str. Ames
Escherichia coli K12
Campylobacter jejuni subsp. *jejuni* NCTC 11168
Mycobacterium tuberculosis H37Rv
Bacillus thuringiensis str. Al Hakam
Clostridium tetani E88
Lactobacillus casei ATCC 334

4.2.1.1 Prediction based on the de Wachter model

Originally in the de Wachter paper (DE WACHTER 1981), the probabilities are calculated based on single base frequencies in the analyzed DNA sequence. For an $(AC)_t$ microsatellite, the probability to find it at a certain position in a sequence is given by the following formula:

$$P(AC)_t = (P_A * P_C)^t * (1 - P_A P_C)^2$$

where t is the number of repeats, P_A is the proportion of A's in the sequence and P_C is the proportion of C's. The second term corrects for the adjacent dinucleotides at each side

of the microsatellite which should not be AC. To take into account the slight increase in probability to find the repeat at the beginning and near the end of the sequence, de Wachter adds a correction term to the basic formula (shown in the Appendix, page 98 of his paper). However, the sequences analyzed here are very large (i.e. >100000), thus including a correction term to correct the start and end of sequence probabilities would not produce significant change in the results. Therefore, I performed my calculations without the correction factor. The de Wachter formula also includes a factorial term which should account for the order of nucleotides in the repeat sequence, or "sequence isomers". A more appropriate way to account for the order of the bases in the repeat, and to consider also higher order biases that are known to occur within DNA sequences, like dinucleotide and trinucleotide biases (see GOLDMAN 1993; see KARLIN and BURGE 1995), is to use dinucleotide and trinucleotide frequencies instead of single nucleotide frequencies to calculate the probabilities. This approach was also taken in some other papers using the de Wachter model (DIERINGER and SCHLÖTTERER 2003; FIELD and WILLS 1998; METZGAR *et al.* 2000; RAJENDRAKUMAR *et al.* 2007).

The modified de Wachter formula used here would look like this for the example $(AC)_t$ dinucleotide:

$$P(AC)_t = (AC/N)^t * N$$

where *AC* is the number of AC motifs in the sequence, *N* is the total length of the sequence, and *t* is the number of repeats of the microsatellite. Then, for all dinucleotides with *t* number of repeats, the expected number would be:

$$X_t = \sum_{x=1}^{i} P(M)^t * N$$

where *i* is the number of all different dinucleotide combinations, *P(M)* is the proportion of a dinucleotide combination in the sequence, *N* and *t* have the same meaning as above.

4.2.1.2 Second order Markov model prediction

The Markov property implies that, given the present state, future states are independent of the past states. The description of the present state fully captures all the information that could influence the future evolution of the process. Future stages are reached through a probabilistic process instead of a deterministic one (KOSKI 2001). To account for

microsatellites with motifs 1 to 3 nucleotides in length, I applied a second order Markov model, calculating the expected frequencies for each motif type as follows:

$$P_{(A)} = \frac{A}{A+T+C+G}$$

$$P_{(A\to C)} = \frac{AC}{AC+AT+AG+AA}$$

$$P_{(AA\to C)} = \frac{AAC}{AAC+AAT+AAG+AAA}$$

where *A, T, C, G, AC, AG*... and so on, represent the number of the respective motifs. observed in the query sequence. $P_{(A\to C)}$ is the probability of having a C given that the last nucleotide was an A, and $P_{(AA\to C)}$ is the probability of having a C given that the last pair of nucleotides were A and A.

Once having the proportions for every possible motif, the expected number of a specific microsatellite, for example $(AC)_3$ was obtained as follows:

$$P_{(ACACAC)} = P_{(A)} * P_{(A\to C)} * P_{(AC\to A)} * \ P_{(CA\to C)} * P_{(AC\to A)} * P_{(CA\to C)}$$

A graphical example of this Markov calculation with respect to the de Wachter calculation can be observed in **figure 4.1**. This process was repeated for all nucleotide combinations from 1 to 3 nucleotides in length, with 2 to 20 repetitions for A/T mononucleotides, 2 to 15 repetitions for C/G mononucleotides, and with 2 to 10 repetitions for di- and trinucleotide repeats.

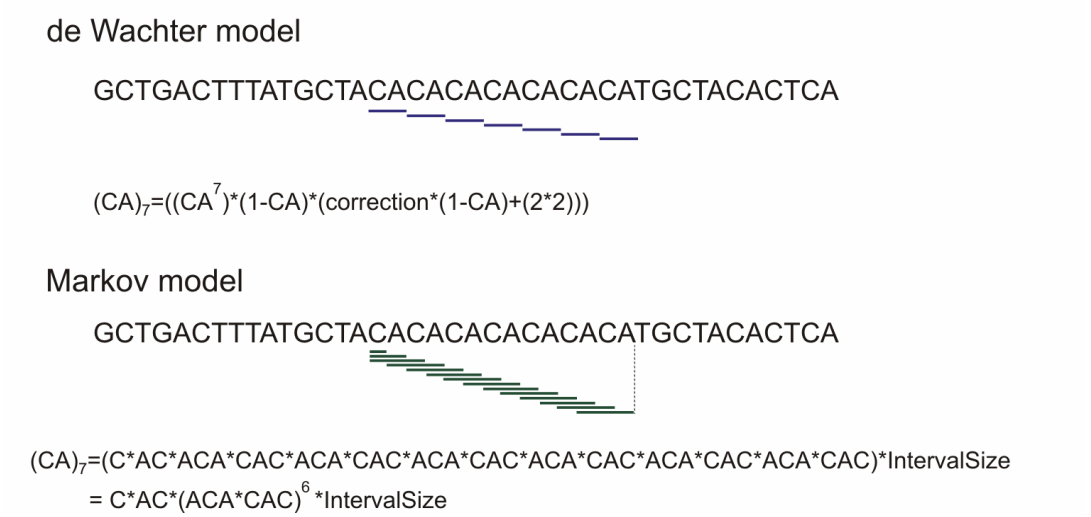

Figure 4.1: Graphical comparison of the calculations for the de Wachter and Markov models used for the prediction of expected microsatellite numbers. For simplicity, the probability of occurrence of a motif CA, written as *P(CA)* for the de Wachter model and as ($P_{C\rightarrow A}$) for the Markov model, are both denoted as CA here. The meaning of CA is the same for both models, but the values are differently calculated.

4.2.2 Observed number of microsatellites

For the comparisons with the expected numbers of microsatellites calculated with the above models it was necessary to count the number of occurrences of perfect tandem repeats independently for each microsatellite motif tested. This specific task could not be performed with any of the available perfect tandem repeat finders reviewed in Chapter II because the kind of redundancy required in the output in this case (hits for each motif reported independently) is usually filtered out by the programs. An option could have been to perform independent searches for each motif with a program which uses a 'dictionary approach', like TROLL (CASTELO *et al.* 2002) but this program does not do exhaustive searches and it has problems processing gaps within sequences (see Chapter II). Therefore, a perfect tandem repeat finding algorithm was designed in collaboration with Carsten Horn (HORN and VARGAS JENTZSCH 2009), who then wrote a version in Delphi 2007. The program reports perfect tandem repetitions exhaustively for all possible permutations of microsatellite motifs up to a given motif length. The minimum number of repeats necessary to report a hit can also be specified, and was set to two repeats. The results are obtained in a coordinates file, with the position of each individual hit, and a summary file, where counts for every motif and the number of repetitions are divided into 100000 nt interval windows to analyze the variation of the microsatellite frequencies within the chromosome.

The microsatellite motifs analyzed were all motif permutations from one to three nt in length (i.e. 4 mononucleotides, 4^2 dinucleotides, 4^3 trinucleotides), except for same nucleotide runs, amounting to 76 different motifs, and the minimum number of repeats to report was set to two. Only integer repetitions were considered (i.e. no incomplete motifs were counted as part of the repeat), and the length of the repeats is therefore reported in number of repeat units.

All calculations were performed in the program R (http://www.r-project.org/), aided by the program Tinn-R (FARIA *et al.* 2001) for writing and organizing the scripts, and visualized in Microsoft Excel spreadsheets.

4.2.3 Comparisons among modeled and observed frequencies

Due to the exponential nature of the data and the diverse and irregular distributions of the observed frequencies, I did not find an appropriate method to statistically test the differences among the generated datasets. To keep the comparisons consistent and comparable between the different genomes, the differences among datasets were measured in percentage deviation of the observed data from either of the models. The degree of deviation from expected frequencies required for a motif to become overrepresented ([obs-exp)/obs]*100) was set at 50%, which would be equivalent to an observed/expected (O/E) ratio of 2. All the frequencies were measured and calculated for chromosomes divided into 100000 nt intervals. To obtain genome-specific minimum length thresholds, the whole-chromosome frequencies for all motifs corresponding to the main repeat types (mono- di-, and trinucleotides) were averaged, and the mode value of the number of repeats surpassing by 50% the expectations for each repeat type and model was taken among the chromosomes of each genome.

4.3 Results and Discussion

The complete genomes of 24 eukaryotes, 8 bacteria, and 5 achaea were scanned for occurrences of mono-, di- and trinucleotide perfect tandem repeats with a minimum of two tandem repetitions. These observed frequencies of tandem repeats were compared with expected values calculated using two models, the de Wachter model and a second order Markov model. The results were graphed in representation plots on a semilogarithmic scale per motif and per chromosome, and were subsequently summarized into four motif type categories: A/T, C/G, dinucleotides, and trinucleotides. Mononucleotides for A/T and C/G motifs were analyzed separately because there were consistently large differences among the frequencies of these repeats within chromosomes.

4.3.1 Differences among expectation models

The expectations obtained with the two models applied here varied among motifs and among genomes. **Figure 4.2** shows a set of representation plots for all dinucleotide combinations in the human chromosome 1. Because the graphs are semilogarithmic, the exponential decrease of the numbers of microsatellite occurrence with respect to microsatellite length (number of repetitions) follows a negative straight line (one line per model). The curves corresponding to the observed numbers of microsatellites show initially an exponential decrease, but these diverge from the straight expectation curves at some point. This divergence point from the observed curve, either from the Markov or from the de Wachter curve, represents the "threshold of overrepresentation".

The threshold of overrepresentation varied based on the model taken into account; for some motifs the expectations were very similar among both models but, in the majority of cases, differences up to 400-fold could be observed among expectations for the range of 2 to 10 repeats (which is the critical range for setting the minimum length threshold in most microsatellites) between the two models. The expectations for each of the 76 different motif combinations from mono- to trinucleotides were calculated separately because Hrabcova and Kypr (2003) had noticed dramatic differences among microsatellite frequencies in human and mouse di- and trinucleotides depending on the starting base (e.g. differences among equivalent motifs CA and AC). These differences were also evident here; it can be observed in **figure 4.2** that the differences among Markov and de Wachter expectation curves were very similar for some motifs while differing largely for others. This variation

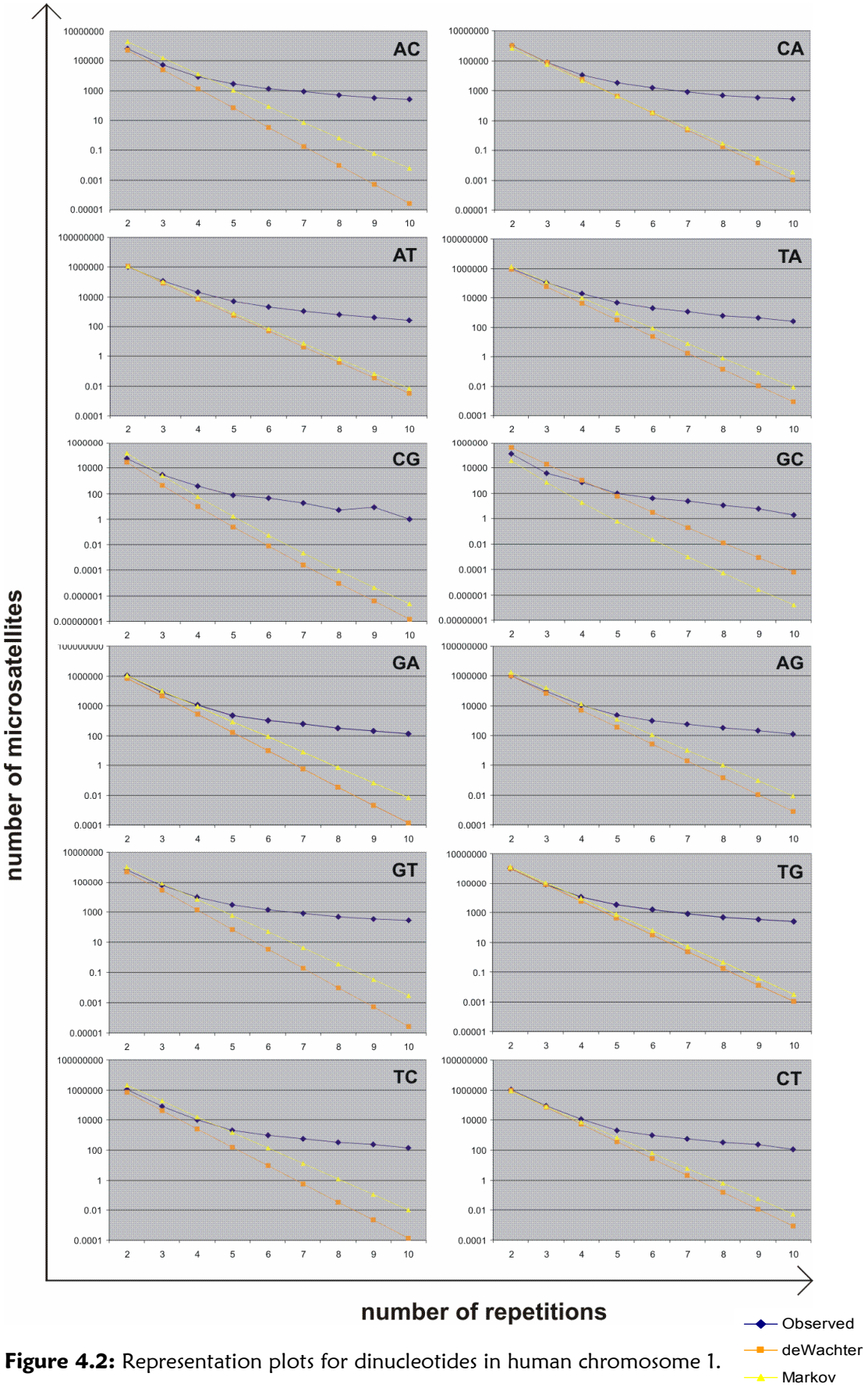

Figure 4.2: Representation plots for dinucleotides in human chromosome 1.

occured regardless of motif self-complementarity; i.e. the expectations for equivalent motifs can differ strongly both with the same model and with different models.

In general, the Markov expectation values were much higher than the de Wachter ones, and this relationship became inverted for most CG-rich motifs (e.g. GC, CCT, CCG). The differences among overrepresentation thresholds were more conspicuous among mono and dinucleotide repeats because the range of possible overrepresentation values is much higher for these motifs. A summary of the chromosomal mode of the minimum length threshold in number of repeats for the four motif categories in all eukaryotic genomes analyzed can be observed in **table 4.3**. Similar summary tables for bacteria and archaea are presented in **tables 4.4** and **4.5**.

Table 4.3: Minimum microsatellite length thresholds in numbers of repeats for all eukaryotic genomes analyzed: mode among all chromosomes for each genome, pooled for AT-rich mononucleotides (A/T), CG-rich mononucleotides (C/G), all dinucleotides (DI), and all trinucleotides(TRI).

Genome	Markov				deWachter			
	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
Homo sapiens	13	12	5	3	7	10	4	3
Pan troglodytes	13	12	5	3	7	10	4	4
Macaca mulatta	13	12	5	4	7	10	4	3
Canis familiaris	12	11	5	4	7	5	4	4
Mus musculus	12	10	5	4	8	8	4	3
Ratus norvergicus	12	10	5	4	8	8	4	4
Equus caballus	13	12	5	4	7	9	4	3
Bos taurus	12	12	5	4	8	10	4	4
Gallus gallus	12	11	5	4	7	8	5	3
Monodelphis domestica	11	10	5	4	8	5	4	5
Ornithorhynchus anatinus	11	12	5	3	8	7	4	3
Arabidopsis thaliana	13	10	5	3	8	10	4	3
Oryza sativa (japonica cultivar-group)	13	10	4	3	7	9	4	3
Danio rerio	13	10	5	3	7	6	4	3
Tetraodon nigroviridis	12	9	5	3	7	7	4	3
Oryzias latipes	15	9	5	3	6	6	4	3
Gasterosteus aculeatus	12	8	5	3	7	6	4	3
Apis mellifera	13	9	5	3	8	5	4	3
Anopheles gambiae str. PEST	13	10	4	3	6	9	4	3
Drosophila melanogaster	12	10	4	3	5	9	4	3
Tribolium castaneum	>20	10	6	3	6	9	6	4
Saccharomyces cerevisiae	12	9	5	3	7	9	5	3
Caenorhabditis elegans	>20	9	5	3	5	9	5	3
Plasmodium falciparum	15	7	8	3	9	4	4	3

Table 4.4: Minimum microsatellite length thresholds in numbers of repeats for all bacterial genomes analyzed

Genome		Markov				deWachter			
		A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
Clostridium tetani	1	NA	NA	4	NA	NA	5	4	NA
	2§	NA	NA	NA	5	NA	5	5	2
Bacillus thuringiensis	1	NA	NA	2	2	NA	NA	2	2
	2§	NA	NA	NA	NA	NA	NA	NA	NA
Brucella melitensis	1	2	2	2	2	2	2	2	2
	2	2	2	2	2	2	2	2	2
Lactobacillus casei		NA	9	NA	5	5	NA	NA	3
Neisseria meningitidis		NA	9	6	3	NA	11	5	2
Escherichia coli K12		NA	NA	NA	3	5	NA	6	3
Campylobacter jejuni		NA	3	NA	3	5	3	NA	NA
Mycobacterium tuberculosis		3	NA	NA	NA	3	NA	NA	NA

NA means that no tandem repeats of the corresponding motif are overrepresented.
§ these are plasmid sequences

Table 4.5: Minimum microsatellite length thresholds in numbers of repeats for all archaeal genomes analyzed

Genome	Markov				deWachter			
	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
Methanosaeta thermophila	10	NA	NA	3	10	NA	6	3
Pyrobaculum aerophilum	NA	12	6	3	5	11	5	3
Hyperthermus butylicus	NA	NA	NA	3	NA	NA	NA	3
Methanocaldococcus jannaschii	NA	NA	NA	NA	6	4	NA	NA
Natronomonas pharaonis	NA	2	NA	2	NA	2	NA	2

NA means that no tandem repeats of the corresponding motif are overrepresented.

The minimum length thresholds for overrepresentation were relatively similar among all eukaryotic genomes analyzed, an exception being the genomes of *Tribolium castaneum* and *Caenorhabditis elegans*. It is for these two species that the expectations among the two models showed the highest divergence. According to the Markov model, poly A/T mononucleotides with less than 20 repetitions are not overrepresented in either genome, while according to the de Wachter model A/T mononucleotides become overrepresented already at 6 and 5 repetitions for *Tribolium castaneum* and *Caenorhabditis elegans* respectively. The minimum length threshold for mononucleotides in *C. elegans* have been studied before using a zero-order Markov model (DECHERING *et al.* 1998) and the de Wachter model (DIERINGER and SCHLÖTTERER 2003). The results from both of these studies contrast with the ones obtained here, with the reported thresholds of 8 and 3 (DECHERING *et al.* 1998)

and 3 and 10 (DIERINGER and SCHLÖTTERER 2003) for poly-A/T and poly-C/G, respectively. The differences from the first study could be due to the zero-order in comparison to the second-order Markov model used here. However, the results from Dieringer and Schlötterer (2003), who used the same de Wachter model, are more difficult to explain; probably the fact that they based the choice of the minimum length thresholds on permutation tests could play a role here.

In bacterial and archaeal genomes the minimum length thresholds were highly variable (**tables 4.4** and **4.5)**, and in most bacteria A/T mononucleotides did not show overrepresentation. An exception was *Mycobacterium tuberculosis,* for which only A/T mononucleotides were significantly overrepresented above 3 repetitions. In the genome of *Brucella melitensis* almost all motif types were overrepresented starting from two repetitions; di- and trinucleotide poly-pyrimidines became overrepresented only above 4 to 6 repetitions, and the trinucleotide motifs CGT and TCG did not show overrepresentation, but overall these threshold values contrast with the other bacteria analyzed for which the minimum length thresholds were higher and most motifs did not become overrepresented (compare **figures 4.3** and **4.4**).

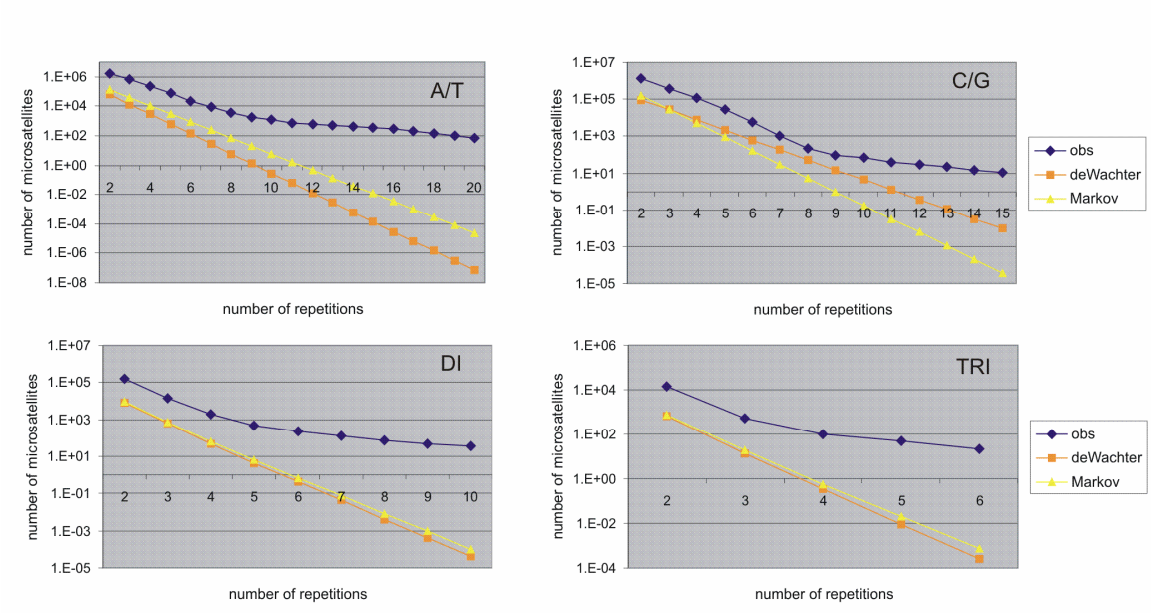

Figure 4.3: Comparison of observed and expected tandem repeat frequencies in the chromosome 1 of *Brucella melitensis*. In this genome tandem repeats of most motifs are overrepresented, even whith only two repetitions.

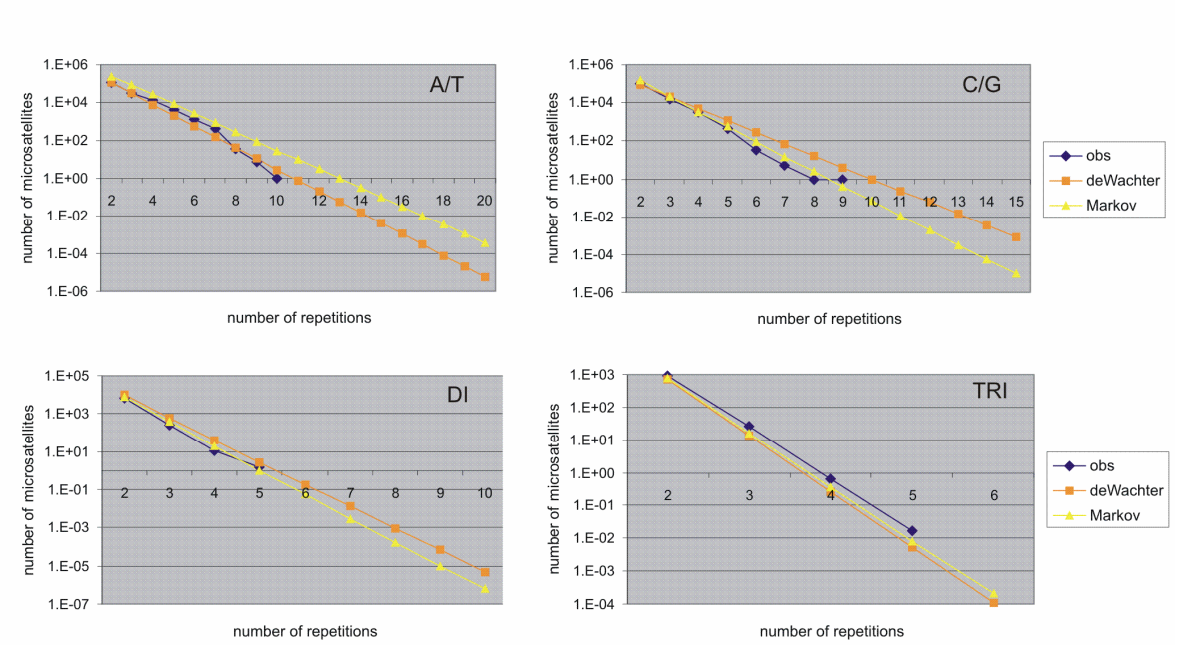

Figure 4.4: Comparison of observed and expected tandem repeat frequencies in the genome of *Lactobacillus casei*. Only C/G mononucleotides with more than 9 repeats and trinucleotides longer than 5 repeats show a weak but significant overrepresentation based on the Markov model.

The appropriateness of the applied expectation models can be tested by simulating DNA sequences based on each of the models, and then counting the occurrence of microsatellites on the simulated sequences. If 1000 simmulations are performed for a query sequence, by chance 5% of these generated sequences should contain some tandem repeats with lengths surpassing the threshold predicted by the model. A model satisfying these criteria should be able to represent the analyzed DNA sequence properly (personal communication Prof. Drik Metzler). The generation of simulated sequences based on the prediction model utilized is also useful to generate a variance of the expectation values, allowing in this way to test statistically whether observed microsatellite frequencies differ significantly from expectations with the characteristic nucleotide compositions (DIERINGER and SCHLÖTTERER 2003; TAUTZ *et al.* 1986): However, the majority of the studies which calculated microsatellite expectations did not perform simulations, probably because the overrepresentation thresholds could easily be drawn from the representation plots (COENYE and VANDAMME 2005; FIELD and WILLS 1998; METZGAR *et al.* 2000; PUPKO and GRAUR 1999; RAJENDRAKUMAR *et al.* 2007; ROSE and FALUSH 1998). Nevertheless, the proposed variance estimation from simulations has a relatively weak statistical value, and it is a better practice to fit observed microsatellite counts to a Poisson distribution for long microsatellites, and to

a normal distribution for the small ones (personal communication Prof. Drik Metzler, MRÀZEK 2006)

Selecting an appropriate model is problematic because real DNA sequences have been shapen by complex processes throughout evolutionary history, like duplications, insertion and deletion of DNA fragments, recombinations, selection, etc. For this reason, homogeneous random models like the ones used here can not represent accurately the compositional heterogeneity of DNA sequences (MRÀZEK 2006). Genomes are influenced by selective forces at both, global and local scales. Therefore, a series of heterogeneous random models have been proposed to attempt at reproducing the compositional characteristics of real DNA seqeunes. For these heterogeneous models, DNA sequences are partitioned into regions of differential composition (coding, non-coding, intergenic, etc.), and the local nucleotide compositions from these regions are used to generate each of them independently. Furthermore, codon frequencies, position of the nucleotide in the codon, and even tetranucleotide composition of coding versus non-coding regions can be included in these models (MRÀZEK 2006). Nevertheless, the aim of the tests in this Chapter was to test if it is justifiable to use the same minimum length threshold for microsatellite searches across different genomes. Therefore, a homogeneous model was expected to provide the necessary information without producing excessive noise.

The two models used here present a good balance between information content and simplicity. Both take into account characteristic mono-, di- and trinucleotide frequencies for each genome, which based on the caos game representation theory are enough to reproduce the majority of patterns observed within DNA sequences. (GOLDMAN 1993). The main difference among these models is the interdependence among nucleotides assumend by the Markov model. The applied second-order Markov model takes into account, for each position in DNA, the preceding two nucleotides in the direction of replication of DNA, and this interdependence assumption was found to represent most DNA sequences appropriately (BLAISDELL 1985).

For the genomes analyzed here, the Markov model predicted higher minimum length thresholds than the de Wachter model, except for most CG-rich motifs with consecutive Cs and Gs (e.g. GC, CCT, CCG). This suggests that that Markov model is more sensible to changes in sequence composition (i.e. CG content) than the de Wachter model. Also, the strong differences among mononucleotide predictions are due to the use of single nucleotide frequencies for the mononucleotide repeat predictions in the de Wachter model,

in contrast to the use of trinucleotide expected frequencies for all motif types in the Markov model. Since Markov models are expected to represent more closely biological systems than completely stochastic models (BEAUMONT and RANNALA 2004), and the differences of the Markov model with the simpler de Wachter model were substantial, I decided to base the estimation of minimum length thresholds on the expectations from the Markov model.

4.3.2 Variation of microsatellite minimum length thresholds based on the second order Markov model

Regardless of the model, tandem repeats with more than 3 to 5 repetitions for di- and trinucleotides, and more than 10 to 13 repetitions in mononucleotides are usually overrepresented within eukaryotic genomes. This is consistent with other studies that used both de Wachter and Markov models to analyze mono- to trinucleotide repeats (DECHERING *et al.* 1998; DIERINGER and SCHLÖTTERER 2003; LAI and SUN 2003).

It became clear from the comparisons of **tables 4.3**, **4.4**, and **4.5**, that eukaryotic genomes tend to contain more overrepresented short tandem repeats than prokaryotic and archaeal genomes. The minimum length thresholds were also very similar among eukaryotes (**figure 4.5**), while in prokaryotes and archaea these vary considerably (**figure 4.6**). The analyses of prokaryotic minimum length thresholds carried out here are only preliminary and were done to complement the eukaryotic analysis. A higher number of bacterial and archaeal genomes would be necessary to draw conclusions about bacterial and archaeal minimum length thresholds. However, it is noticeable that most of the bacterial and archaeal genomes analyzed here have mostly trinucleotide repeats overrepresented in their genomes, and in cases like *Methanocaldococcus jannaschii* .and *Bacillus thuringiensis*, there are probably no microsatellites in these genomes. In contrast to my restuls, Field and Willis (FIELD and WILLS 1998) found that *E. coli* and *M. jannaschii* have an overrepresentation of mononucleotides starting at two repeats, based on a de Wachter model. Gur-Arie *et al.* (2000), studied mono to hexanucleotides in *E. coli*, and in comparison with random computer generated genomes based on single nucleotide permutations, concluded that only mono- and trinucleotide repeats are overrepresented in the *E.coli* genome. The predictions from the de Wachter model used here are consistent with the results from Gur Aire *et al.* (2000), although dinucleotides higher than 6 repeats were also overrepresented. The threshold of 50% deviation which I chose for my analyses is comparatively high, and therefore weak overrepresentations will probably not reach this

threshold. Due to this conservative approach, the thresholds shown in **tables 4.3** and **4.4** represent most likely microsatellites which are"active" within genomes.

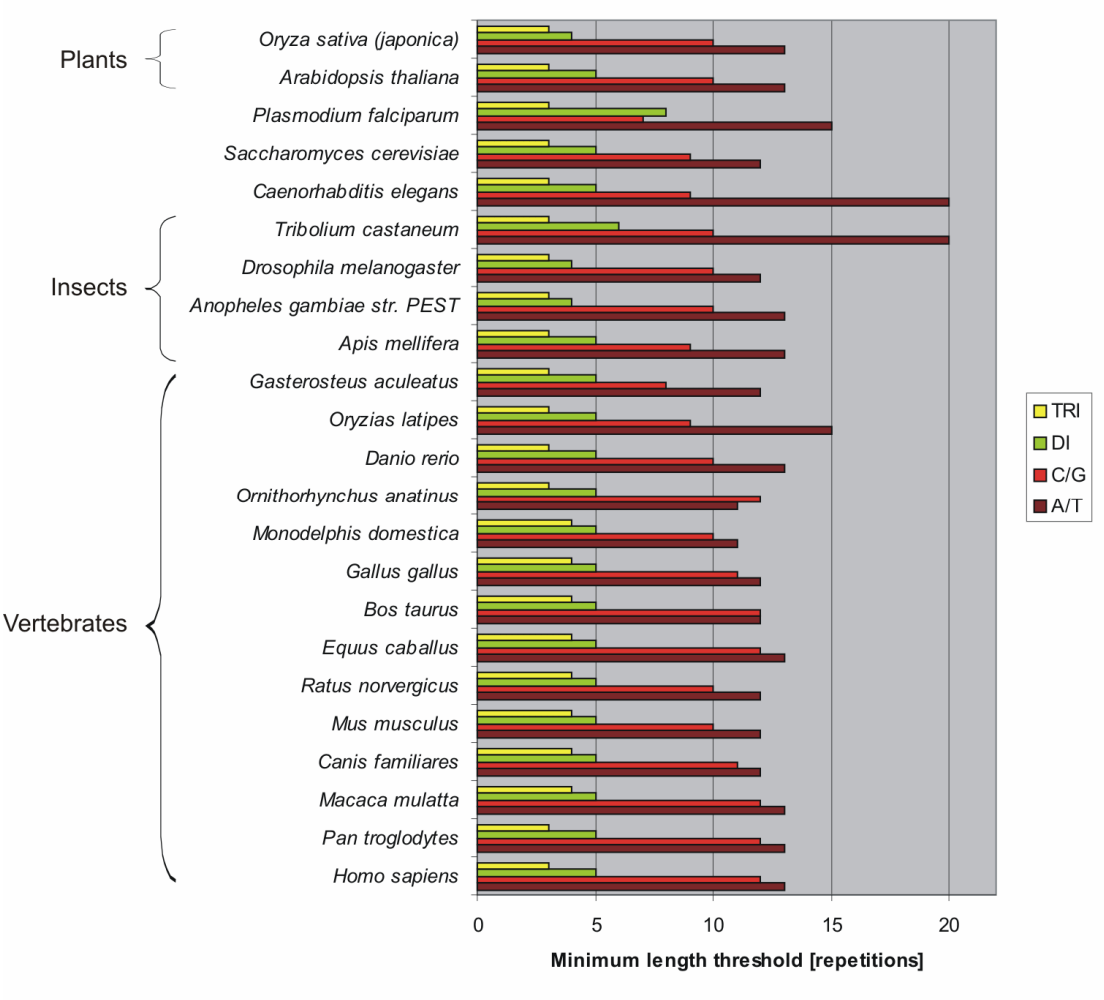

Figure 4.5: Comparison of minimum length thresholds among eukaryotes based on the Markov model expectations.

Despite the apparent homogeneity in minimum length threshold values among eukaryotic genomes, these showed variable degrees of motif overrepresentation, and therefore also variable minimum length thresholds among chromosomes. The least variation between chromosomes and among genomes in general is observed among the vertebrate species (mammalian, avian, fish) analyzed here. The minimum length mode summaries per chromosome for all genomes analyzed can be observed in **tables A2** to **A25** in the appendix section. The highest variation exists among chromosomes in the yeast genome, with AT-rich tandem repeats strongly overrepresented, and G/C mononucleotides practically absent from 6 of the 16 chromosomes. One chromosome in the opossum

genome, chromosome X, differed from the rest in that all motifs are overrepresented starting at two repeats. All other chromosomes showed a constant pattern of minimum thresholds: A/T 11, C/G 10, DI 5 and TRI 4, based on the Markov model.

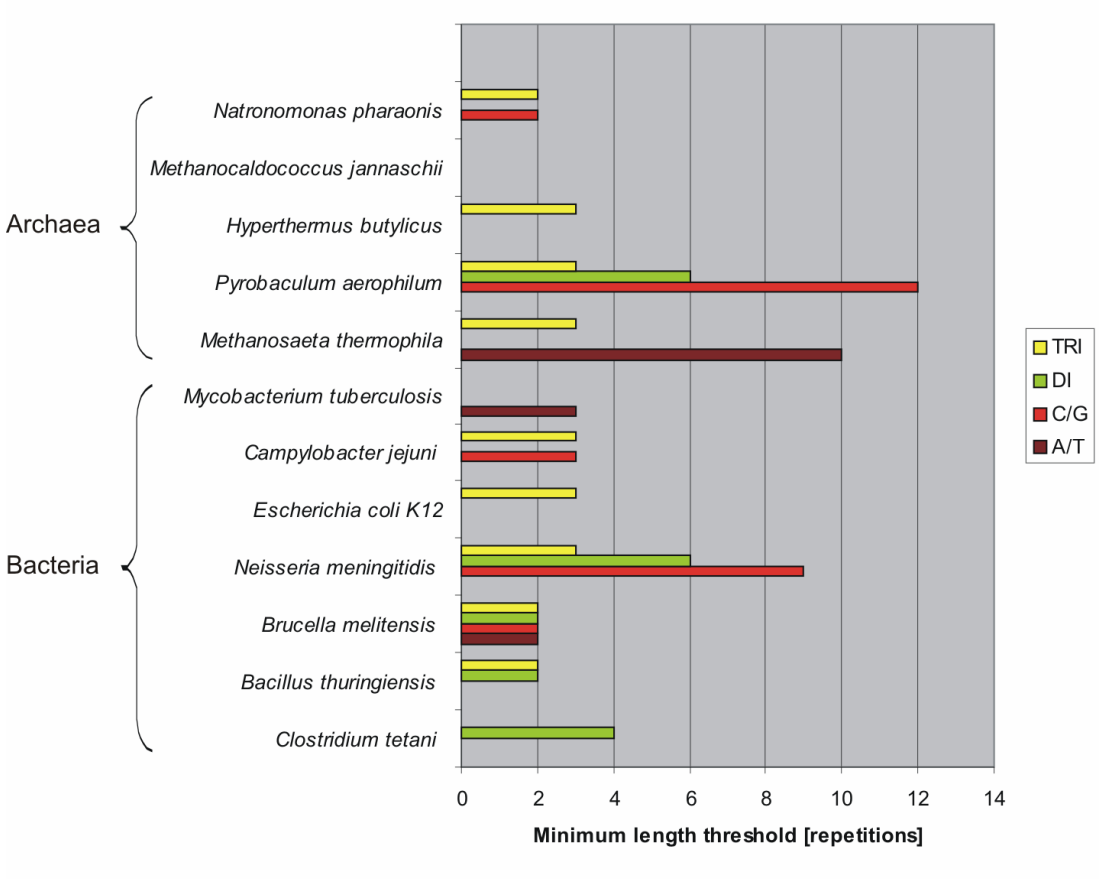

Figure 4.6: Comparison of minimum length thresholds among prokaryotes and archaea based on the Markov model expectations.

Rose and Falush (1998), Pupko and Graur (1999), and Dieringer and Schlötterer (2003), who also performed comparisons of observed and expected microsatellite frequencies in eukaryotic genomes argued that the minimum microsatellite length can be better analyzed when expressing it in absolute numbers of nucleotides instead of numbers of repetitions. This is because the relationship of microsatellite length vs the ratio of observed to expected frequencies on a semi-logarithmic scale is similar for different motifs when expressing the length in nucleotide numbers. Expressing the microsatellite lengths in numbers of repeats makes the differences more pronounced (see **figure 4.6**). It is expected that comparisons among different motifs based on numbers of repeats will be biased because different motifs have different lengths. Expressing the length of the microsatellites in numbers of

nucleotides is a way to standardize the distributions to make them comparable. However, the clean parallel relationship shown in the first graph in **figure 4.5** is more characteristic of mammalian and other higher eukaryotes. Species with smaller genomes and more variable microsatellite minimum length thresholds, like *Plasmodium* or *C. elegans* may show less defined relationships among motifs, as shown in **figure 4.6**, and the differences among motifs become accented when the graphs are constructed based on the Markov expectations. For the purpose of defining microsatellite searches, expressing the microsatellite length in absolute length may introduce bias and uncertainty in the analysis (see discussion in Chapter II, page 85). The number of repetitions in a microsatellite, however, is a relative measure which refers directly to the main mutational units of microsatellites. Therefore, even if the overrepresentation thresholds would amount to equivalent values in numbers of nucleotides, it is still my recommendation that this length be translated into numbers of microsatellites for the purpose of defining microsatellites.

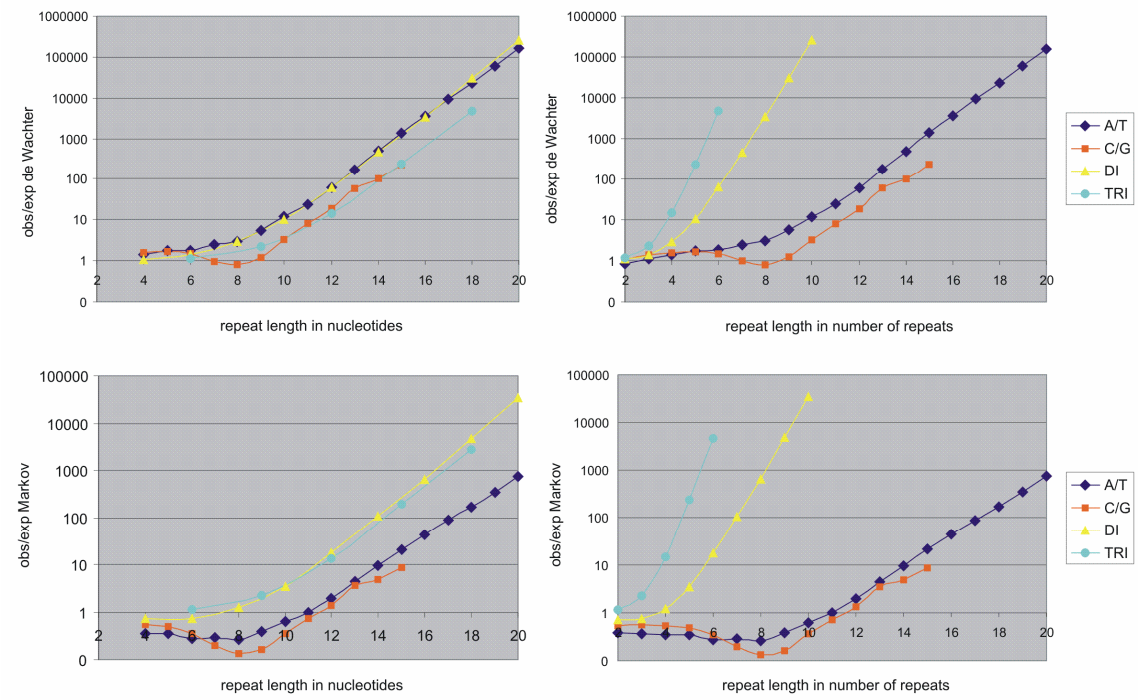

Figure 4.7: Comparison of observed and expected tandem repeat frequencies in the human chromosome 1. The upper graphs correspond to O/E ratios calculated based on de Wachter expected value, and the bottom graphs correspond to the Markov expectations. In the left graphs, the length of microsatellites is expressed in numbers of nucleotides, and the right graphs are based on numbers of repeats. The differences among motif types are stronger based on the Markov expectations in comparison to the de Wachter expectations.

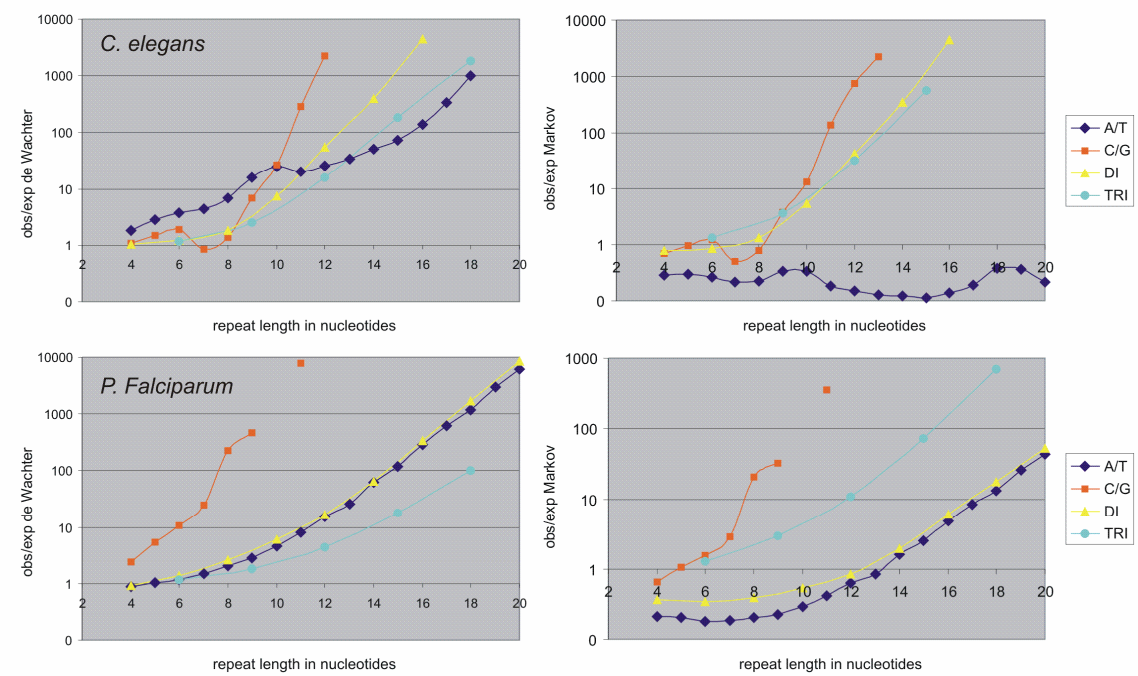

Figure 4.8: O/E ratios vs repeat length in nucleotides based on the de Wachter (left) and Markov (right) models. The motif types for the two species depicted here, *C. elegans* and *P. falciparum*, do not show the parallel relationship seen for the human and yeast genomes in previous publications (DIERINGER and SCHLÖTTERER 2003; PUPKO and GRAUR 1999).

The experiments and calculations presented in this chapter were performed under the hypothesis that the mutational properties of microsatellites, i.e. elevated mutation rates and the tendency to increase in length, will bring about a net increase in microsatellite abundance across the sequence. Therefore, all tandem repeats of microsatellite motifs would be likely to show genome-wide tendencies towards overrepresentation. This is an oversimplification of reality and rests on the assumptions that microsatellites mutate by replication slippage, that the dominant factors governing microsatellite mutation are directly affected by sequence composition of the microsatellite motif (specifically nucleotide, dinucleotide, and trinucleotide proportions), and that these factors will have similar effects in different genome regions and in all taxa. However, the results presented here provide a baseline to test if the same microsatellite length threshold can be expected, and therefore used, in comparisons across different genomes, this in order to make microsatellite abundance data more comparable among these.

4.4 Conclusions

The most influential parameter for microsatellite searches and for defining a microsatellite locus itself based on its genomic context is the 'minimum length threshold'.

Based on both, a second order Markov model, and on the stochastic de Wachter model (DE WACHTER 1981), the minimum length threshold for short tandem repeats to become overrepresented in DNA sequences varies among motifs and motif types within and between genomes.

The Markov model predicts, in general, higher minimum length thresholds than the de Wachter model. The differences among models are more conspicuous in mononucleotides, where the Markov model predicts more variation than the de Wachter model (minimum thresholds from 7 to 13 based on the Markov model, and from 6 to 10 repeats based on the de Wachter model). Therefore, the Markov model seems to be more sensible to changes in sequence composition.

When motifs are categorized into AT-rich mononucleotides, CG-rich mononucleotides, dinucleotides, and trinucleotides, the mode thresholds are very similar among all vertebrates, with a slight reduction in the thresholds of aquatic vertebrates. Therefore, for inter-species comparisons of microsatellite abundance, a global threshold of 12, 5, and 4 repeats for mono-, di-, and trinucleotides, could be used for the non-aquatic vertebrates. Aquatic vertebrates show higher differences among AT- and CG-rich mononucleotides. Therefore, it would be appropriate to use separate thresholds for these mononucleotides: 13, 9, 5, 3 repeats for AT- and CG-mononucleotides, di-, and trinucleotides, respectively.

There are strong differences in minimum microsatellite length thresholds between eukaryotes, prokaryotes, and archaea. Therefore, it is recommendable to use species-specific minimum length thresholds to carry out microsatellite searches and comparisons between these groups. Prokaryotes and archaea have much lower minimum length thresholds for mono- to trinucleotides than eukaryotes, and in many cases microsatellites with as few as two repeats would already be overrepresented. This confirms that lower thresholds for microsatellite searches should be used for prokaryotes and archaea.

The second-order Markov model used here takes into account, for each position in DNA, the preceding two nucleotides in the direction of replication of DNA. This model showed the highest differences among species, especially between more distant taxa. Therefore, I

recommend using this model for the verification of minimum length thresholds for microsatellite motifs in genomes, before embarking in microsatellite abundance and distribution studies.

4.5 References

ARMOUR, J. A., R. NEUMANN, S. GOBERT and A. J. JEFFREYS, 1994 Isolation of human simple repeat loci by hybridization selection. Hum Mol Genet **3:** 599-565.

BACHTROG, D., S. WEISS, B. ZANGERL, G. BREM and C. SCHLÖTTERER, 1999 Distribution of dinucleotide microsatellites in the *Drosophila melanogaster* genome. Mol Biol Evol **16:** 602-610.

BACOLLA, A., J. E. LARSON, J. R. COLLINS, J. LI, A. MILOSAVLJEVIC *et al.*, 2008 Abundance and length of simple repeats in vertebrate genomes are determined by their structural properties. Genome Res **18:** 1545-1553.

BEAUMONT, M. A., and B. RANNALA, 2004 The Bayesian revolution in genetics. Nat Rev Genet **5:** 251-261.

BLAISDELL, B. E., 1985 Markov chain analysis finds a significant influence of neighboring bases on the occurrence of a base in eucaryotic nuclear DNA sequences both protein-coding and noncoding. J Mol Evol **21:** 278-288.

BROHEDE, J., N. ARNHEIM and H. ELLEGREN, 2004 Single-molecule analysis of the hypermutable tetranucleotide repeat locus D21S1245 through sperm genotyping: a heterogeneous pattern of mutation but no clear male age effect. Mol Biol Evol **21:** 58-64.

CASTELO, A. T., W. MARTINS and G. R. GAO, 2002 TROLL--tandem repeat occurrence locator. Bioinformatics **18:** 634-636.

COENYE, T., and P. VANDAMME, 2005 Characterization of mononucleotide repeats in sequenced prokaryotic genomes. DNA Res **12:** 221-233.

COX, R., and S. M. MIRKIN, 1997 Characteristic enrichment of DNA repeats in different genomes. Proc Natl Acad Sci U S A **94:** 5237-5242.

DE WACHTER, R., 1981 The number of repeats expected in random nucleic acid sequences and found in genes. J Theor Biol **91:** 71-98.

DECHERING, K. J., K. CUELENAERE, R. N. KONINGS and J. A. LEUNISSEN, 1998 Distinct frequency-distributions of homopolymeric DNA tracts in different genomes. Nucleic Acids Res **26:** 4056-4062.

DIERINGER, D., and C. SCHLÖTTERER, 2003 Two distinct modes of microsatellite mutation processes: evidence from the complete genomic sequences of nine species. Genome Res **13:** 2242-2251.

DOMANIC, N. O., and F. P. PREPARATA, 2007 A novel approach to the detection of genomic approximate tandem repeats in the Levenshtein metric. J Comput Biol **14:** 873-891.

EDWARDS, Y. J., G. ELGAR, M. S. CLARK and M. J. BISHOP, 1998 The identification and characterization of microsatellites in the compact genome of the Japanese pufferfish, *Fugu rubripes*: perspectives in functional and comparative genomic analyses. J Mol Biol **278:** 843-854.

ELLEGREN, H., 2004 Microsatellites: simple sequences with complex evolution. Nat Rev Genet **5:** 435-445.

FARIA, J. C., P. GROSJEAN and E. JELIHOVSCHI, 2001 Tinn-R GUI editor for R language and environment, pp.

FIELD, D., and C. WILLS, 1998 Abundant microsatellite polymorphism in *Saccharomyces cerevisiae*, and the different distributions of microsatellites in eight prokaryotes

and *S. cerevisiae*, result from strong mutation pressures and a variety of selective forces. Proc Natl Acad Sci U S A **95:** 1647-1652.

FUJIMORI, S., T. WASHIO, K. HIGO, Y. OHTOMO, K. MURAKAMI *et al.*, 2003 A novel feature of microsatellites in plants: a distribution gradient along the direction of transcription. FEBS Lett **554:** 17-22.

GENTLES, A. J., and S. KARLIN, 2001 Genome-scale compositional comparisons in eukaryotes. Genome Res **11:** 540-546.

GOLDMAN, N., 1993 Nucleotide, dinucleotide and trinucleotide frequencies explain patterns observed in chaos game representations of DNA sequences. Nucleic Acids Res **21:** 2487-2491.

GUO, W. J., J. LING and P. LI, 2009 Consensus features of microsatellite distribution: Microsatellite contents are universally correlated with recombination rates and are preferentially depressed by centromeres in multicellular eukaryotic genomes. Genomics.

GUR-ARIE, R., C. J. COHEN, Y. EITAN, L. SHELEF, E. M. HALLERMAN *et al.*, 2000 Simple sequence repeats in *Escherichia coli*: abundance, distribution, composition, and polymorphism. Genome Res **10:** 62-71.

HORN, C., and I. M. VARGAS JENTZSCH, 2009 IrSa perfect tandem repeat finder, pp. Gloon Systems, Moenchengladbach.

HRABCOVA, I., and J. KYPR, 2003 Genomic occurrence of microsatellites containing integral and non-integral repeat numbers. Biochem Biophys Res Commun **300:** 824-831.

JURKA, J., and C. PETHIYAGODA, 1995 Simple repetitive DNA sequences from primates: compilation and analysis. J Mol Evol **40:** 120-126.

KARLIN, S., and C. BURGE, 1995 Dinucleotide relative abundance extremes: a genomic signature. Trends Genet **11:** 283-290.

KOSKI, T., 2001 *Hidden Markov models for bioinformatics*. Kluwer Academic Publishers Norwell, Massachusetts.

LAI, Y., and F. SUN, 2003 The relationship between microsatellite slippage mutation rate and the number of repeat units. Mol Biol Evol **20:** 2123-2131.

LEVINSON, G., and G. A. GUTMAN, 1987 Slipped-strand mispairing: a major mechanism for DNA sequence evolution. Mol Biol Evol **4:** 203-221.

LIM, S., L. NOTLEY-MCROBB, M. LIM and D. A. CARTER, 2004 A comparison of the nature and abundance of microsatellites in 14 fungal genomes. Fungal Genet Biol **41:** 1025-1036.

MARX, K. A., S. T. HESS and R. D. BLAKE, 1993 Characteristics of the large (dA).(dT) homopolymer tracts in *D. discoideum* gene flanking and intron sequences. J Biomol Struct Dyn **11:** 57-66.

MESSIER, W., S. H. LI and C. B. STEWART, 1996 The birth of microsatellites. Nature **381:** 483.

METZGAR, D., J. BYTOF and C. WILLS, 2000 Selection against frameshift mutations limits microsatellite expansion in coding DNA. Genome Res **10:** 72-80.

MORGANTE, M., M. HANAFEY and W. POWELL, 2002 Microsatellites are preferentially associated with nonrepetitive DNA in plant genomes. Nat Genet **30:** 194-200.

MRÀZEK, J., 2006 Analysis of distribution indicates diverse functions of simple sequence repeats in Mycoplasma genomes. Mol Biol Evol **23:** 1370-1385.

PRASAD, M. D., M. MUTHULAKSHMI, M. MADHU, S. ARCHAK, K. MITA *et al.*, 2005 Survey and analysis of microsatellites in the silkworm, *Bombyx mori*: frequency, distribution, mutations, marker potential and their conservation in heterologous species. Genetics **169:** 197-214.

PRIMMER, C. R., A. P. MOLLER and H. ELLEGREN, 1996 A wide-range survey of cross-species microsatellite amplification in birds. Mol Ecol **5:** 365-378.

PUPKO, T., and D. GRAUR, 1999 Evolution of microsatellites in the yeast *Saccharomyces cerevisiae*: role of length and number of repeated units. J Mol Evol **48:** 313-316.

RAJENDRAKUMAR, P., A. K. BISWAL, S. M. BALACHANDRAN, K. SRINIVASARAO and R. M. SUNDARAM, 2007 Simple sequence repeats in organellar genomes of rice: frequency and distribution in genic and intergenic regions. Bioinformatics **23:** 1-4.

RICHARD, G. F., and F. PAQUES, 2000 Mini- and microsatellite expansions: the recombination connection. EMBO Rep **1:** 122-126.

ROSE, O., and D. FALUSH, 1998 A threshold size for microsatellite expansion. Mol Biol Evol **15:** 613-615.

SUBRAMANIAN, S., R. K. MISHRA and L. SINGH, 2003 Genome-wide analysis of microsatellite repeats in humans: their abundance and density in specific genomic regions. Genome Biol **4:** R13.

TAUTZ, D., M. TRICK and G. A. DOVER, 1986 Cryptic simplicity in DNA is a major source of genetic variation. Nature **322:** 652-656.

TOTH, G., Z. GASPARI and J. JURKA, 2000 Microsatellites in different eukaryotic genomes: survey and analysis. Genome Res. **10:** 967-981.

VALDES, A. M., M. SLATKIN and N. B. FREIMER, 1993 Allele frequencies at microsatellite loci: the stepwise mutation model revisited. Genetics **133:** 737-749.

VIGOUROUX, Y., Y. MATSUOKA and J. DOEBLEY, 2003 Directional evolution for microsatellite size in maize. Mol Biol Evol **20:** 1480-1483.

WEBSTER, M. T., N. G. C. SMITH and H. ELLEGREN, 2002 Microsatellite evolution inferred from human-chimpanzee genomic sequence alignments. Proceedings of the National Academy of Sciences of the United States of America **99:** 8748-8753.

WIERDL, M., M. DOMINSKA and T. D. PETES, 1997 Microsatellite instability in yeast: dependence on the length of the microsatellite. Genetics **146:** 769-779.

XU, X., M. PENG and Z. FANG, 2000 The direction of microsatellite mutations is dependent upon allele length. Nat Genet **24:** 396-399.

ZHU, Y., J. E. STRASSMANN and D. C. QUELLER, 2000 Insertions, substitutions, and the origin of microsatellites. Genetical Research **76:** 227-236.

General Discussion

I showed throughout Chapters II to IV that that the computer-based identification of microsatellites is an intricate process that is affected by multiple factors and can therefore produce a wide range of non-biology related variation in the results. Mainly due to these seemingly understated sources of variation, it is difficult, and at times not possible, to perform valid comparisons among microsatellite abundance and distribution studies. These comparability problems arise either because different programs with uncharacterized microsatellite finding abilities are commonly used, or because different definitions of microsatellites, differing mainly in the minimum length thresholds and the proportion of imperfection allowed within hits, are applied in different studies.

One of the main sources of trouble is that the publications and instructions for repeat finding programs do not usually include a characterization of the effects of program parameters on the results. The end-user of a program is supposed to know better than the programmer what is required from the program in a specific situation and, therefore, it is this end-user who is expected to optimize the program's parameters. However, usually, biologists take this testing and optimizing for granted and seldom move beyond the program's default parameters, therefore limiting their analyses to a set of microsatellites mainly defined mathematically and without consideration of biologically relevant parameters. If the aim of the repeat search is to use the resulting hits for the development of molecular markers, or other tasks where exhaustive search results are not imperative, most available tools will prove useful. However, in cases where the results of the microsatellite search are to be used for genomic structure characterization and inter-genomic comparisons of microsatellites, and any task herein, the choice of an appropriate repeat finding program, and the further optimization of its search parameters, becomes crucial.

The testing and characterization of a program's behaviour, however, is a task which would best be fulfilled by the authors of the respective programs themselves. The authors will know better which parameters are most important for the search and how to interpret the variation observed in the results. Otherwise, the user needs to go through a process of re-discovery, like the one carried out throughout Chapters II and III of this thesis, to make sure that the program will indeed produce the expected results. The ideal situation for the development of microsatellite identification tools, as well as any other bioinformatic tool,

would be to have both, biologists as well as informaticians, working together in a group. In this way the produced tools can be tested and troubleshouted not only based on informatics parameters, but also taking into account biologically relevant characteristics. It is also of critical importance that information on both, advantages and drawbacks of application programs, gets published together with the program so that the tools can be used and developed further in an efficient way.

I presented here a characterization of search parameters for the programs TRF and SciRoKo showing that the differences among these program's results are mainly due to the parameters chosen, rather than algorithm-specific limitations, as is sometimes implied in publications which present program comparisons (LECLERCQ *et al.* 2007; MUDUNURI and NAGARAJARAM 2007; WEXLER *et al.* 2005). By constructing microsatellite number and coverage distributions, as the ones presented in Chapter III, the parameter values with which different programs produce comparable output can be assessed, to subsequently perform in-depth analyses by performing intersection or subtraction operations on the output files. Furthermore, for the comparison of microsatellite search results it is also important to go beyond the simple comparison of numbers of hits obtained. This is essential when dealing with imperfect microsatellite searches because, in this case, the number of hits reported by a program will depend on the program's algorithm as well as on the degree of imperfection and clustering of microsatellites within the query sequences. Therefore, I recommend using both, the number of hits as well as the coverage or total length of microsatellite hits for reporting microsatellite abundance and distribution results. In this way the results will be comparable among studies which use different programs, and which describe microsatellite content in different genomes.

One of the most influential parameters in microsatellite searches is the minimum length above which a tandem repeat hit can be considered as a microsatellite, because the number of microsatellites detected increases exponentially when decreasing the microsatellite length (DECHERING *et al.* 1998). Microsatellites are defined as being overrepresented throughout genomes due to their high mutation rates and a general tendency towards expansion (XU *et al.* 2000). In this sense, it has been proposed that microsatellites become overrepresented within genomes only above a specific length threshold, because a certain number of repeats are necessary for the repeat array to be prone to strand slippage replication and other microsatellite mutation mechanisms (LAI and SUN 2003; MESSIER *et al.* 1996; ROSE and FALUSH 1998). Below this threshold, the probability of strand slippage is expected to be very low, while above the threshold slippage mutations will dominate over

point mutations. Although there is an ongoing discussion on the existence and the exact value of this minimum length threshold (see DIERINGER and SCHLÖTTERER 2003; LAI and SUN 2003; see PUPKO and GRAUR 1999; RAJENDRAKUMAR *et al.* 2007), and several empirical studies have been published on the subject (SREENU *et al.* 2006; ZHU *et al.* 2000), the majority of studies of microsatellite abundance and distribution use arbitrary values for the minimum length threshold (see Introduction in Chapter IV, page 137).

Usually, a single minimum length threshold value is used for all microsatellite motif sizes, like for example a threshold of 11 nt used by Fujimori *et al.* (2003) to analyze microsatellites in *Oryza sativa*, *Arabidopsis thaliana*, *Homo sapiens*, and *Mus musculus*. Otherwise, individual thresholds for each motif size can be used as in the case of Katti *et al.* (2001): 20, 10, 7, and 5 repeats for mono, di, tri, and tetranucleotides, respectively, for the comparison of microsatellite distribution between *Homo sapiens*, *Drosophila melanogaster*, *Caenorhabditis elegans*, *Arabidopsis thaliana* and *Sacharomyces cerevisiae*. In either case, the use of the same minimum length threshold among different, and sometimes highly divergent, genomes, resides on the assumption of neutrality of microsatellite mutations. Based on a neutral model of microsatellite evolution, microsatellite abundance will vary according to nucleotide composition and higher order compositional biases of the sequence, and the minimum length threshold for microsatellite motif sizes shouldn't significantly change (BELL and JURKA 1997; KRUGLYAK *et al.* 1998). To test the appropriateness of using the same minimum length threshold for the detection of microsatellites in all genomes, I applied two random models: a second order Markov model and a combinatoric model published by de Wachter (1981) to calculate expectations of microsatellite occurrence for motifs from 1 to 3 nt. The overrepresentation thresholds calculated with each of these models differed mainly for mononucleotide repeat types and for CG-rich motifs. The Markov model expectations were then used to estimate the minimum length thresholds for microsatellites in prokaryotic, eukaryotic and archaeal genomes. For both models, the minimum length thresholds were similar for all vertebrates, and diverged slightly for the rest of the eukaryotes analyzed. In contrast, the thresholds calculated for bacterial and archaeal genomes showed considerable variation. As taxa diverge, it is likely that the specific factors affecting microsatellite dynamics will diverge too, and therefore comparisons may become biased if species-specific thresholds are not used. Therefore, it seems appropriate to use species-specific minimum length thresholds for microsatellite abundance comparisons, if possible, for every new genomic sequence released.

Based on the methodology designed throughout Chapters II, III, and IV, I performed a preliminary analysis of microsatellite abundance across the complete and assembled genomic sequences of 24 eukaryotes, 8 prokaryotes, and 5 archaea (results in Appendix III). Microsatellites were detected using TRF for imperfect repeats and SciRoKo in MISA mode for perfect microsatellites. Also, two different sets of minimum length threshold were used: the species-specific minimum length thresholds estimated in Chapter IV (see supplemental methods in Appendix II), and the standard minimum length threshold corresponding to the most commonly used minimum length threshold in the literature (see EDWARDS *et al.* 1998; GUO *et al.* 2009; JURKA and PETHIYAGODA 1995; MORGANTE *et al.* 2002; SUBRAMANIAN *et al.* 2003; TÓTH *et al.* 2000): 12 nt, used in terms of repeats as 12, 6, 4 3, 3, and 3 repeats for mono- to hexanucleotide motifs respectively. The results of this preliminary analysis confirmed that the use of species-specific minimum lengtht thresholds can produce significantly different microsatellite datasets, mainly in genomes highly divergent from vertebrates. This is because the standard minimum length threshold of 12 nt represents more closely the minimum length threshold sets obtained for vertebrates using the second order Markov model. From the non-vertebrate eukaryotic genomes analyzed, the only ones which strongly diverged from the standard minimum length threshold were *C. elegans* and *T. castaneum*, mainly due to very large minimum length thresholds for A/T mononucleotides.

The strongest differences among minimum length thresholds were observed in prokaryotic and archaeal genomes. The microsatellite abundance in these genomes was also significantly lower than in eukaryotes, in agreement with reports from earlier studies (COENYE and VANDAMME 2005; KASSAI-JÁGER *et al.* 2008). This was a very heterogeneous group of sequences, and a larger sample of genomes from both taxa would be needed to see if more defined groups can be found. This relatively small sample of genomes was included in this study as a means of basal comparison for the eukaryotic genomes. It became evident that a larger variability in microsatellite evolutionary stages can be observed in prokaryotes and archaea, and that it is not adequate to use the same definition of microsatellite (mainly in terms of minimum length threshold) to analyze these genomes.

It is debatable if thresholds based on overrepresentation, like the ones calculated and used in this thesis and in other papers (e.g. DECHERING *et al.* 1998; DIERINGER and SCHLÖTTERER 2003; LAI and SUN 2003), provide enough evidence to decide if a microsatellite hit should be filtered out of a dataset or not. The thresholds for A/T mononucleotides based on the Markov model are relatively high, mostly above 10 to 12 repeats, and tandem repeats shorter that that could well be hypermutable based on in-vivo observations (KARAOGLU *et al.*

2005; NYEO and YU 2007; RAJENDRAKUMAR *et al.* 2007). However, as the species-specific minimum length thresholds used here are based on empirical overrepresentation of repeats in each genome, the microsatellite datasets presented represent the fraction of microsatellites likely to be "active" within genomes. In other words, these are microsatellites with a high potential of mutation, because the increased mutation rates of microsatellites are the most likely factors driving the overrepresentation of these sequences within genomes (LAI and SUN 2003). Moreover, the differences in the overrepresentation of microsatellite loci among different genomes also suggest the existence of different microsatellite mutation dynamics among genomes.

The analyses presented throughout this thesis challenge the classic concept of microsatellites as simple and independent repeated regions which mutate randomly throughout genomes. Several pieces of evidence have been accumulated showing that microsatellite mutation processes are neither predictable nor generalizable. First, microsatellite abundance varies throughout genomes, and this variation is independent of genomic size (CRUZ *et al.* 2005; EDWARDS *et al.* 1998; LIM *et al.* 2004; MORGANTE *et al.* 2002) and nucleotide composition (LAI and SUN 2003; NDIFON *et al.* 2006), suggesting that these are not completely random sequences. Second, perfect microsatellite repeats are usually immersed within longer imperfect repeat stretches where the original motif can still be recognized, which constitute an important phase of microsatellite evolution (TAUTZ *et al.* 1986), the extent of which has until now been poorly characterized. Third, the tandemly repeated structure of microsatellites, which is highly prone to insertion-deletion mutations and to rearrangements via recombination processes, constitutes a highly versatile source of variation with potential for exaptation (METZGAR and WILLS 2000). For these reasons, I think it is important to redefine the way microsatellite abundance and distribution analyzes throughout genomes are performed.

The methodologies and suggestions presented throughout this thesis aim at improving the comparability among studies on microsatellite abundance. For this I worked through several steps involved in the definition of microsatellites. It may be argued that the specific details of microsatellite definition and analysis do not necessarily need to be standardized and followed throughout studies in order to advance our knowledge about microsatellite and genome evolution. After all, strong patterns and other specific features about sequences as conspicuous as microsatellites are bound to come to light regardless of the specific definitions used for the studies (Christian Schlötterer, personal communication). On the other hand, at the current pace of data generation, which is expected to increase both

quantitatively (i.e. the number of genomes for different taxa as well as several genome versions for the same species) and qualitatively (thanks to new generation sequencing technologies), I think it is important to start accumulating mutually compatible data on microsatellites. A good example to justify the need of unambiguous definitions and appropriate is the Genomic Standards Consortium (http://gensc.wordpress.com/), a project where major international efforts are being put in place to standardize the computer and biological language used for defining genomic data. The obvious benefits are easier access to complete and accurate data and the possibility to perform meta-analyses when enough data becomes available.

References

BELL, G. I., and J. JURKA, 1997 The length distribution of perfect dimer repetitive DNA is consistent with its evolution by an unbiased single-step mutation process. J Mol Evol **44:** 414-421.

COENYE, T., and P. VANDAMME, 2005 Characterization of mononucleotide repeats in sequenced prokaryotic genomes. DNA Res **12:** 221-233.

CRUZ, F., M. PEREZ and P. PRESA, 2005 Distribution and abundance of microsatellites in the genome of bivalves. Gene **346:** 241-247.

DE WACHTER, R., 1981 The number of repeats expected in random nucleic acid sequences and found in genes. J Theor Biol **91:** 71-98.

DECHERING, K. J., K. CUELENAERE, R. N. KONINGS and J. A. LEUNISSEN, 1998 Distinct frequency-distributions of homopolymeric DNA tracts in different genomes. Nucleic Acids Res **26:** 4056-4062.

DIERINGER, D., and C. SCHLÖTTERER, 2003 Two distinct modes of microsatellite mutation processes: evidence from the complete genomic sequences of nine species. Genome Res **13:** 2242-2251.

EDWARDS, Y. J., G. ELGAR, M. S. CLARK and M. J. BISHOP, 1998 The identification and characterization of microsatellites in the compact genome of the Japanese pufferfish, *Fugu rubripes*: perspectives in functional and comparative genomic analyses. J Mol Biol **278:** 843-854.

FUJIMORI, S., T. WASHIO, K. HIGO, Y. OHTOMO, K. MURAKAMI *et al.*, 2003 A novel feature of microsatellites in plants: a distribution gradient along the direction of transcription. FEBS Lett **554:** 17-22.

GUO, W. J., J. LING and P. LI, 2009 Consensus features of microsatellite distribution: Microsatellite contents are universally correlated with recombination rates and are preferentially depressed by centromeres in multicellular eukaryotic genomes. Genomics.

JURKA, J., and C. PETHIYAGODA, 1995 Simple repetitive DNA sequences from primates: compilation and analysis. J Mol Evol **40:** 120-126.

KARAOGLU, H., C. M. LEE and W. MEYER, 2005 Survey of simple sequence repeats in completed fungal genomes. Mol Biol Evol **22:** 639-649.

KASSAI-JÁGER, E., C. ORTUTAY, G. TÓTH, T. VELLAI and Z. GASPARI, 2008 Distribution and evolution of short tandem repeats in closely related bacterial genomes. Gene **410:** 18-25.

KATTI, M. V., P. K. RANJEKAR and V. S. GUPTA, 2001 Differential distribution of simple sequence repeats in eukaryotic genome sequences. Mol Biol Evol **18:** 1161-1167.

KRUGLYAK, S., R. T. DURRETT, M. D. SCHUG and C. F. AQUADRO, 1998 Equilibrium distributions of microsatellite repeat length resulting from a balance between slippage events and point mutations. Proceedings of the National Academy of Sciences of the United States of America **95:** 10774-10778.

LAI, Y., and F. SUN, 2003 The relationship between microsatellite slippage mutation rate and the number of repeat units. Mol Biol Evol **20:** 2123-2131.

LECLERCQ, S., E. RIVALS and P. JARNE, 2007 Detecting microsatellites within genomes: significant variation among algorithms. BMC Bioinformatics **8:** 125.

LIM, S., L. NOTLEY-MCROBB, M. LIM and D. A. CARTER, 2004 A comparison of the nature and abundance of microsatellites in 14 fungal genomes. Fungal Genet Biol **41:** 1025-1036.

MESSIER, W., S. H. LI and C. B. STEWART, 1996 The birth of microsatellites. Nature **381:** 483.

METZGAR, D., and C. WILLS, 2000 Evidence for the adaptive evolution of mutation rates. Cell **101:** 581-584.

MORGANTE, M., M. HANAFEY and W. POWELL, 2002 Microsatellites are preferentially associated with nonrepetitive DNA in plant genomes. Nat Genet **30:** 194-200.

MUDUNURI, S. B., and H. A. NAGARAJARAM, 2007 IMEx: Imperfect Microsatellite Extractor. Bioinformatics **23:** 1181-1187.

NDIFON, W., A. NKWANTA and D. HILL, 2006 Some probabilistic results on the nonrandomness of simple sequence repeats in DNA sequences. Bulletin of Mathematical Biology **68:** 1747-1759.

NYEO, S.-L., and J.-P. YU, 2007 Length distributions of simple tandem repeats in genomes. Journal of Biological Systems **15:** 299-312.

PUPKO, T., and D. GRAUR, 1999 Evolution of microsatellites in the yeast *Saccharomyces cerevisiae*: role of length and number of repeated units. J Mol Evol **48:** 313-316.

RAJENDRAKUMAR, P., A. K. BISWAL, S. M. BALACHANDRAN, K. SRINIVASARAO and R. M. SUNDARAM, 2007 Simple sequence repeats in organellar genomes of rice: frequency and distribution in genic and intergenic regions. Bioinformatics **23:** 1-4.

ROSE, O., and D. FALUSH, 1998 A threshold size for microsatellite expansion. Mol Biol Evol **15:** 613-615.

SREENU, V. B., P. KUMAR, J. NAGARAJU and H. A. NAGARAJARAM, 2006 Microsatellite polymorphism across the *M. tuberculosis* and *M. bovis* genomes: implications on genome evolution and plasticity. BMC Genomics **7:** 78.

SUBRAMANIAN, S., R. K. MISHRA and L. SINGH, 2003 Genome-wide analysis of microsatellite repeats in humans: their abundance and density in specific genomic regions. Genome Biol **4:** R13.

TAUTZ, D., M. TRICK and G. A. DOVER, 1986 Cryptic simplicity in DNA is a major source of genetic variation. Nature **322:** 652-656.

TÓTH, G., Z. GASPARI and J. JURKA, 2000 Microsatellites in different eukaryotic genomes: survey and analysis. Genome Res. **10:** 967-981.

WEXLER, Y., Z. YAKHINI, Y. KASHI and D. GEIGER, 2005 Finding approximate tandem repeats in genomic sequences. J Comput Biol **12:** 928-942.

XU, X., M. PENG and Z. FANG, 2000 The direction of microsatellite mutations is dependent upon allele length. Nat Genet **24:** 396-399.

ZHU, Y., J. E. STRASSMANN and D. C. QUELLER, 2000 Insertions, substitutions, and the origin of microsatellites. Genetical Research **76:** 227-236.

Appendix I : Additional Figures and Tables

Table A1: Input parameters for tandem repeat finders analyzed in Chapter II

Program	Year	Publication	min unit length	max unit length	points match	mismatch penalty	indel penalty	min score	min length	min number of repetitions	imperfection
Tandyman	1997	*NP LEACH and CLELAND 1997	-l (minim m 2) ?	-u	na	na	na	na	na	-m	na
Tandem Repeat Finder (TRF)	1999	BENSON 1999	na	Maxperiod, 7th value	1st value	2nd value	3rd value	6th value	min score/match points	na	implied in match and indel probabilities (PM and PI)
SSR screener	2000	*NP GUR-ARIE et al. 2000	asked for	max10	na	na	na	na	asked for	asked for	na
SSRIT Simple Sequence Repeat Identification Tool	2001	*NP TEMNYKH et al. 2001	by modification in the perl code		na	na	na	na	na	by modification in the perl code	na
Tandem Repeat Occurrence Locator (TROLL)	2002	CASTELO et al. 2002	na	-M (motifs specified in motif file)	na	na	na	na	-m	na	na
MIcroSAtellite identification tool (MISA)	2002	*NP Author: Thomas Thiel THIEL et al. 2003	na	na	na	na	na	na	specified for each motif size in auxiliary ini file		'interruptions' in ini file

Table A1: Input parameters for tandem repeat finders (continued)

Program	Year	Publication	min unit length	max unit length	points match	mismatch penalty	indel penalty	min score	min length	min number of repetitions	imperfection
Sputnik II	2003	*NP LA ROTA et al. 2005	-u	-v	-m	-n	na	-s	-L	na	-r (maximum recursion) -R
STRING	2003	PARISI et al. 2003						Fifth value during input			
			na	na	na	na	na		na	na	na
mreps	2003	KOLPAKOV et al. 2003	- minperiod	-maxperiod					-minsize	-exp	-res (resolution)
					na	na	na	na			
Perfect Tandem Repeat Finding Executable	2003	application paper: COLLINS et al. 2003		-repsiz						-minrep	
					na	na	na	na	na		na
Approximate Tandem Repeats hunter (ATRhunter)	2004	WEXLER et al. 2005		maximum motif length	1st value	2nd value	3rd value	linked with minimum similarity level			minimum similarity level
			na						na	na	
Search for Tandem Approximate Repeats (STAR)	2004	DELGRANGE and RIVALS 2004	motif list specified in motif file								
					na	na	na	na	na	na	na
Tandem Repeat Analyzer (TRA E-TRA)	2004	BILGEN et al. 2004	minimum motif length	maximum motif length						can be specified individually for each motif, or as one value for all.	
					na	na	na	Score	na		na

Table A1: Input parameters for tandem repeat finders (continued)

Program	Year	Publication	min unit length	max unit length	points match	mismatch penalty	indel penalty	min score	min length	min number of repetitions	imperfection
Msatfinder Msatminer	2005	*NP THURSTON and FIELD 2005	motif threshold (default 1 to 6)		na	na	na	na	na	can be specified for each motif	na
Phobos	2006	MAYER 2007	-u	-U	Value fixed to 1	-m	-g	-minScore	-l	na	-r
Imperfect Microsatelltie Extractor (IMEx 1.0)	2007	MUDUNURI and NAGARAJARAM 2007	na	na	na	na	na	na	na	n' value per motif size	max imperfection ('k'), imperfection percentage ('p')
Tandem Repeat Software (TRED)	2007	SOKOL et al. 2007	MIN_ PERIOD	MAX_ PERIOD	na	ERROR_VAL		MIN_ RATING	MIN_ LENGTH	na	MAX_ERRORS
SciRoKo (SciRoKoCo)	2007	KOFLER et al. 2007	na (default 1)	na (default 6)	na	-p	na	-s	-l -seedl -seedr	-r, -m (for misa mode)	-mmao
tandem	2007	DOMANIC and PREPARATA 2007	min repeat period	max repeat period (up to 500)	alignment score parameters			minimum score	na	na	probability of matches and indels

*NP : No scientific journal publication was available describing the algorithm. Therefore I cite directly the authors and/or the application paper where the program was first used.

Table A2: Intra-genomic variation of minimum length thresholds in the human genome.

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	247249719	13	13	5	3	7	10	4	3
2	242951149	13	12	5	3	7	10	4	3
3	199501827	13	12	5	3	7	10	4	3
4	191273063	13	12	5	3	7	10	4	3
5	180857866	13	12	5	3	7	10	4	3
6	170899992	13	12	5	3	7	9	4	3
7	158821424	13	12	5	3	7	10	4	3
8	146274826	13	12	5	3	7	10	4	3
9	140273252	13	13	5	3	7	10	4	3
10	135374737	13	12	5	3	7	10	4	3
11	134452384	13	13	5	3	7	10	4	3
12	132349534	13	12	5	3	7	10	4	3
13	114142980	13	12	5	3	7	9	4	3
14	106368585	13	13	5	3	7	10	4	3
15	100338915	12	12	5	3	7	10	4	3
16	88827254	12	13	5	3	6	10	4	3
17	78774742	12	13	5	3	5	10	4	3
18	76117153	13	12	5	3	7	9	4	3
19	63811651	12	10	5	3	5	10	4	3
20	62435964	12	13	5	3	7	10	4	3
21	46944323	13	14	5	3	7	10	4	3
22	49691432	12	14	5	3	5	10	4	3
X	154913754	13	12	5	3	7	9	4	3
Y	57772954	13	12	5	4	7	10	4	4

Table A3: Intra-genomic variation of minimum length thresholds in the chimpanzee genome.

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	229974691	13	12	5	3	7	10	4	4
2A	114460064	13	12	5	3	7	9	4	4
2B	248603653	13	12	5	4	7	9	4	4
3	203962478	13	11	5	3	7	9	4	4
4	194897272	13	11	5	4	7	9	4	4
5	183994906	13	11	5	4	7	9	4	4
6	173908612	13	11	5	4	7	9	4	4
7	160261443	13	11	5	4	7	9	4	4
8	145085868	13	11	5	4	7	9	4	4
9	138509991	13	12	5	4	7	9	4	4
10	135001995	13	12	5	3	7	10	4	4
11	134204764	13	12	5	3	7	10	4	4
12	135371336	13	12	5	3	7	9	4	4
13	115868456	13	12	5	3	7	9	4	4
14	107349158	13	12	5	3	7	10	4	3
15	100063422	13	11	5	3	7	9	4	4
16	90682376	12	12	5	4	7	10	4	3
17	83384210	13	12	5	4	7	10	4	3
18	77261746	13	12	5	3	7	10	4	3
19	64473437	12	12	5	3	5	10	4	3
20	62293572	13	12	5	3	7	10	4	3
21	46489110	13	12	5	3	7	10	4	4
22	50165558	12	NA	5	3	5	10	4	3
X	155361357	13	11	5	4	8	9	4	4

Table A4: Intra-genomic variation of minimum length thresholds in the rhesus genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	228252215	13	13	5	3	7	10	4	3
2	189746636	13	12	5	4	7	9	4	4
3	196418989	13	12	5	4	7	10	4	4
4	167655696	13	11	5	4	7	9	4	3
5	182086969	13	11	5	4	8	9	4	4
6	178205221	13	12	5	4	7	9	4	4
7	169801366	13	12	5	3	7	10	4	3
8	147794981	13	12	5	3	7	10	4	3
9	133323859	13	12	5	3	7	10	4	3
10	94855758	12	13	5	3	6	10	4	3
11	134511895	13	12	5	3	7	9	4	3
12	106505843	13	12	5	3	7	10	4	3
13	138028943	13	12	5	4	7	10	4	3
14	133002572	13	12	5	3	7	10	4	3
15	110119387	13	13	5	4	7	10	4	3
16	78773432	13	12	5	3	7	10	4	3
17	94452569	13	11	5	4	8	9	4	4
18	73567989	13	11	5	4	7	9	4	4
19	64391591	13	14	5	4	5	10	4	3
20	88221753	12	13	5	3	7	10	4	3
X	153947521	13	11	5	4	7	9	4	4

Table A5: Intra-genomic variation of minimum length thresholds in the dog genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	125616256	12	11	5	4	7	10	4	4
2	88410189	12	11	5	4	7	9	4	4
3	94715083	13	11	5	4	7	9	4	4
4	91483860	12	11	5	4	7	9	4	4
5	91976430	13	10	5	4	7	9	4	4
6	80642250	12	11	5	4	7	9	4	4
7	83999179	12	11	5	4	7	9	4	4
8	77315194	12	11	5	4	7	9	4	4
9	64418924	12	11	5	4	7	9	4	4
10	72488556	12	11	5	4	7	9	4	4
11	77416458	12	11	5	4	7	10	4	4
12	75515492	13	11	5	4	7	10	4	4
13	66182471	12	11	5	4	7	9	4	4
14	63938239	13	11	5	4	7	9	4	4
15	67211953	13	11	5	4	7	10	4	3
16	62570175	12	11	5	4	7	10	4	3
17	67347617	12	11	5	4	7	9	4	4
18	58872314	12	12	5	4	7	9	4	4
19	56771304	13	10	5	4	7	9	4	4
20	61280721	12	10	5	4	7	10	4	3
21	54024781	12	11	5	4	7	9	4	4
22	64401119	13	10	5	4	7	10	4	3
23	55389570	12	11	5	4	7	9	4	4
24	50763139	12	11	5	4	7	9	4	4
25	54563659	12	11	5	4	7	9	4	4
26	42029645	12	11	5	3	7	10	4	3
27	48908698	12	11	5	4	7	10	4	3
28	44191819	13	10	5	4	7	10	4	3
29	44831629	12	11	5	3	7	10	4	3
30	43206070	12	11	5	4	7	10	4	3
31	42263495	13	10	5	4	7	9	4	4
32	41731424	13	10	5	4	7	9	4	4
33	34424479	12	11	5	4	7	9	4	4
34	45128234	12	11	5	4	7	10	4	3
35	29542582	12	11	5	4	5	10	4	3
36	33840356	12	11	5	4	7	10	4	3
37	33915115	12	11	5	4	7	10	4	4
38	26897727	12	11	5	4	5	10	4	3
X	126883977	13	11	5	4	8	9	4	4

Table A6: Intra-genomic variation of minimum length thresholds in the mouse genome

		Markov				deWachter			
Chr	Chrom. length	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	197069962	12	10	5	4	8	8	4	3
2	181976762	12	10	5	3	8	8	4	3
3	159872112	12	10	5	4	8	8	4	3
4	155029701	12	10	5	4	8	8	4	3
5	152003063	12	10	5	3	8	8	4	3
6	149525685	12	10	5	4	8	8	4	3
7	145134094	12	10	5	4	8	8	4	3
8	132085098	12	10	5	3	8	8	4	3
9	124000669	12	10	5	4	8	8	4	3
10	129959148	12	10	5	3	8	8	4	3
11	121798632	12	10	5	3	7	8	4	3
12	120463159	12	10	5	4	8	8	4	3
13	120614378	12	10	5	4	8	8	4	3
14	123978870	12	10	5	4	8	8	4	3
15	103492577	12	10	5	3	8	8	4	3
16	98252459	12	10	5	4	8	8	4	3
17	95177420	12	10	5	3	8	8	4	3
18	90736837	12	10	5	4	8	8	4	3
19	61321190	12	10	5	3	7	8	4	3
X	165556469	13	9	5	4	8	5	4	4
Y	16029404	13	10	5	4	7	7	4	4

Table A7: Intra-genomic variation of minimum length thresholds in the rat genome

		Markov				deWachter			
Chr	Chrom. length	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	267910886	12	10	5	4	8	9	4	4
2	258207540	12	10	5	4	8	8	4	4
3	171063335	12	10	5	4	8	9	4	4
4	187126005	12	10	5	4	8	8	4	4
5	173096209	12	10	5	4	8	8	4	4
6	147636619	12	10	5	4	8	9	4	4
7	143002779	12	10	5	4	8	9	4	4
8	129041809	12	10	5	3	8	9	4	4
9	113440463	12	10	5	4	8	9	4	4
10	110718848	11	10	5	4	7	9	4	3
11	87759784	12	10	5	4	8	8	4	4
12	46782294	11	11	5	4	7	9	4	3
13	111154910	12	10	5	4	8	8	4	4
14	112194335	12	10	5	4	8	8	4	4
15	109758846	12	10	5	4	8	9	4	4
16	90238779	12	10	5	4	8	9	4	4
17	97296363	12	10	5	4	8	9	4	4
18	87265094	12	10	5	4	8	8	4	4
19	59218465	11	10	5	4	8	9	4	4
20	55268282	12	10	5	4	8	9	4	4
X	160699376	13	10	5	4	9	8	4	4

Table A8: Intra-genomic variation of minimum length thresholds in the horse genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	125616256	13	12	5	3	7	9	4	3
2	88410189	13	12	5	4	7	9	4	3
3	94715083	13	12	5	4	7	9	4	3
4	91483860	13	11	5	4	7	6	4	3
5	91976430	13	12	5	4	7	9	4	3
6	80642250	13	11	5	4	7	9	4	3
7	83999179	13	12	5	4	7	9	4	3
8	77315194	13	12	5	4	7	9	4	3
9	64418924	13	12	5	4	7	9	4	3
10	72488556	13	12	5	4	7	9	4	3
11	77416458	13	12	5	4	7	9	4	4
12	75515492	13	11	5	4	7	9	4	4
13	66182471	13	11	5	4	7	9	4	3
14	63938239	13	12	5	4	7	9	4	3
15	67211953	13	12	5	4	7	9	4	3
16	62570175	13	12	5	4	7	9	4	3
17	67347617	13	12	5	4	7	9	4	3
18	58872314	13	12	5	4	7	6	4	4
19	56771304	13	11	5	4	7	9	4	3
20	61280721	13	11	5	4	7	9	4	4
21	54024781	13	11	5	4	7	9	4	4
22	64401119	13	11	5	4	7	9	4	4
23	55389570	13	11	5	4	7	9	4	4
24	50763139	13	11	5	4	7	9	4	4
25	54563659	13	11	5	4	7	9	4	4
26	42029645	13	12	5	4	7	9	4	4
27	48908698	13	11	5	4	7	6	4	4
28	44191819	13	12	5	4	7	9	4	4
29	44831629	13	11	5	4	7	6	4	4
30	43206070	13	11	5	4	7	9	4	3
31	42263495	13	11	5	4	7	9	4	3
X	126883977	13	11	5	4	7	5	4	4

Table A9: Intra-genomic variation of minimum length thresholds in the cow genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	102834029	13	12	5	4	9	9	4	4
2	86543008	12	>15	5	3	9	10	4	4
3	85360813	12	12	5	4	8	10	4	4
4	69556449	13	12	5	4	9	10	4	4
5	76426644	13	12	5	4	8	9	4	4
6	69624268	12	13	5	4	8	10	4	4
7	69141744	12	13	5	4	8	10	4	4
8	62115791	12	12	5	4	8	9	4	4
9	64650424	12	13	5	4	8	10	4	4
10	70001009	12	12	5	4	8	9	4	4
11	87172399	12	13	5	4	8	10	4	4
12	48608624	12	12	5	4	8	10	4	4
13	62721844	12	12	5	4	8	10	4	4
14	50711261	12	12	5	4	8	10	4	4
15	53815408	12	12	5	4	8	10	4	4
16	56991954	12	13	5	4	8	10	4	4
17	45915461	12	12	5	4	8	10	4	4
18	56508949	12	12	5	4	8	10	4	4
19	56387648	12	12	5	4	8	10	4	4
20	42952723	12	12	5	4	9	9	4	4
21	49724630	12	13	5	4	8	10	4	4
22	48330986	12	13	5	4	8	10	4	4
23	41602928	12	12	5	4	8	9	4	4
24	45245742	12	12	5	4	9	9	4	4
25	41414979	12	12	5	4	8	10	4	4
26	35715000	12	12	5	4	8	10	4	4
27	31566707	12	13	5	4	8	10	4	4
28	34887537	12	12	5	4	9	9	4	4
29	45822729	12	12	5	4	8	9	4	4
X	48857030	12	11	5	4	9	9	4	4

Table A10: Intra-genomic variation of minimum length thresholds in the chicken genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	2.01E+08	12	10	5	4	7	8	5	4
2	1.55E+08	12	10	5	4	7	8	5	4
3	1.14E+08	12	10	5	4	7	8	5	4
4	94230402	12	10	5	4	7	8	5	3
5	62238931	12	10	5	4	7	8	5	3
6	37400442	12	11	5	4	7	8	5	3
7	38384769	12	10	5	4	7	8	5	3
8	30671729	12	11	5	4	7	8	5	3
9	25554352	12	10	5	4	7	8	5	3
10	22556432	12	10	5	4	7	8	5	3
11	21928095	12	11	5	4	7	8	5	3
12	20536687	12	11	5	4	7	8	5	3
13	18911934	12	10	5	4	7	8	5	3
14	15819469	11	11	6	4	7	8	5	3
15	12968165	11	11	5	4	7	8	5	3
17	11182526	12	11	5	4	7	8	5	3
18	10925261	11	12	6	4	7	8	5	3
19	9939723	11	12	6	4	7	8	5	3
20	13986235	11	12	6	4	7	8	5	3
21	6959642	11	11	5	4	7	8	5	3
22	3936574	12	11	5	4	7	8	5	3
25	2031799	12	11	5	4	7	8	5	3
26	5102438	12	11	5	4	7	8	5	3
27	4841970	11	11	5	3	7	8	5	3
28	4512026	11	12	6	4	7	8	5	3
W	259642	12	11	5	4	7	9	5	3
Z	74602320	13	10	5	4	7	8	5	4

Table A11: Intra-genomic variation of minimum length thresholds in the opossum genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	748055161	11	10	5	4	8	5	4	5
2	541556283	11	10	5	4	8	5	4	5
3	527952102	11	10	5	4	8	5	4	5
4	435153693	11	10	5	4	8	5	4	5
5	304825324	11	10	5	4	8	5	4	5
6	292091736	11	9	5	4	8	5	4	5
7	260857928	11	10	5	4	8	5	4	5
8	312544902	11	10	5	4	8	5	4	5
X	79335909	2	2	2	2	2	2	2	2

Table A12: Intra-genomic variation of minimum length thresholds in the platypus genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	47594283	11	12	5	3	8	7	4	3
2	54797317	10	12	5	3	8	7	4	3
3	59581953	11	12	5	3	8	7	4	3
4	58987262	11	12	5	3	8	7	4	3
5	24609220	11	12	5	3	8	8	4	3
6	16302927	11	12	5	3	8	7	4	3
7	40039088	11	12	5	3	8	7	4	3
8	11243762	11	12	5	3	7	8	4	3
9	6809224	11	12	5	3	7	7	4	3
10	15872666	11	12	5	3	7	7	4	3
11	2696122	10	12	5	3	7	5	4	3
12	3786880	10	12	5	3	7	8	4	3
14	1399469	10	13	5	3	7	5	4	3
15	6611290	11	11	5	3	8	8	4	3
16	1816412	11	11	6	3	7	9	4	3
17	45541551	11	12	5	3	8	7	4	3
18	5652501	10	11	5	3	8	8	4	3
20	5951358	11	11	5	3	7	7	4	3
X1	27786739	11	11	5	3	8	6	4	3
X2	47594283	11	12	5	3	8	7	4	3
X3	54797317	10	12	5	3	8	7	4	3
X5	59581953	11	12	5	3	8	7	4	3

Table A13: Intra-genomic variation of minimum length thresholds in the *Arabidopsis thaliana* genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	30432563	13	10	5	3	8	10	4	3
2	19705359	13	10	5	3	8	9	4	3
3	23470805	13	10	5	3	8	10	4	3
4	18585042	12	10	5	3	8	9	4	3
5	26992728	13	9	5	3	8	10	4	3

Table A14: Intra-genomic variation of minimum length thresholds in the rice genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	43261740	13	10	4	3	6	9	4	3
2	35954743	13	10	4	3	6	9	4	3
3	36192742	13	10	4	3	7	9	4	3
4	35498469	12	10	4	3	7	9	4	3
5	29737217	13	10	4	3	7	9	4	3
6	30731886	13	10	4	3	7	9	4	3
7	29644043	13	10	4	3	7	9	4	3
8	28434780	13	10	4	3	7	9	4	3
9	22696651	14	10	4	3	7	9	4	3
10	22685906	13	10	4	3	7	9	4	3
11	28386948	13	10	4	3	7	9	4	3
12	27566993	14	10	4	3	7	9	4	3

Table A15: Intra-genomic variation of minimum length thresholds in the zebrafish genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	70589895	13	10	5	3	7	6	4	3
2	61889685	13	10	5	3	7	6	4	3
3	77179095	13	11	5	3	7	6	4	3
4	47249802	13	10	5	3	7	6	4	3
5	84656180	13	10	5	3	7	6	4	3
6	69554819	13	10	5	3	7	6	4	3
7	87691871	13	10	5	3	7	6	4	3
8	66798501	13	10	5	3	7	6	4	3
9	55712184	13	10	5	3	7	6	4	3
10	54070595	13	10	5	3	7	6	4	3
11	52342180	13	10	5	3	7	6	4	3
12	58719258	13	9	5	3	7	6	4	3
13	64258675	13	9	5	3	7	6	4	3
14	91717235	13	10	5	3	7	6	4	3
15	57214918	13	9	5	3	7	6	4	3
16	65489547	13	9	5	3	7	6	4	3
17	63411520	13	9	5	3	7	6	4	3
18	59765243	13	10	5	3	7	6	4	3
19	51715404	13	9	5	3	7	6	4	3
20	63653707	13	10	5	3	7	6	4	3
21	56255777	13	10	5	3	7	6	4	3
22	47751166	13	10	5	3	7	6	4	3
23	53215897	13	10	5	3	7	6	4	3
24	46081529	13	10	5	3	7	6	4	3
25	40315040	13	9	5	3	7	6	4	3

Table A16: Intra-genomic variation of minimum length thresholds in the medaka genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	28185914	15	9	5	3	7	6	4	3
2	23295652	15	9	5	3	6	6	4	3
3	16798506	15	9	5	3	7	6	4	3
4	32632948	15	9	5	3	7	6	4	3
5	12251397	15	9	5	3	6	6	4	3
6	17083675	15	9	5	3	6	6	4	3
7	27937443	15	9	5	3	6	6	4	3
8	19368704	15	9	5	3	6	6	4	3
9	20249479	15	9	5	3	7	6	4	3
10	15657440	15	8	5	3	7	6	4	3
11	16706052	15	9	5	3	7	6	4	3
12	18401067	15	9	5	3	7	6	4	3
13	20083130	15	9	5	3	7	6	4	3
14	15246461	15	9	5	3	7	6	4	3
15	16198764	15	9	5	3	7	6	4	3
16	18115788	15	9	5	3	7	6	4	3
17	14603141	15	9	5	3	6	6	4	3
18	16282716	15	9	5	3	7	6	4	3
19	20240660	15	9	5	3	6	6	4	3
20	19732071	15	9	5	3	7	6	4	3
21	11717487	15	9	5	3	7	6	4	3

Table A17: Intra-genomic variation of minimum length thresholds in the stickleback genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
I	28185914	12	9	5	3	7	6	4	3
II	23295652	12	8	5	3	7	6	4	3
III	16798506	12	8	5	3	7	6	4	3
IV	32632948	12	9	5	3	7	6	4	3
V	12251397	12	8	5	3	7	6	4	3
VI	17083675	12	8	5	3	7	7	4	3
VII	27937443	12	8	5	3	7	7	4	3
VIII	19368704	12	8	5	3	7	6	4	3
IX	20249479	12	9	5	3	7	6	4	3
X	15657440	12	8	5	3	7	6	4	3
XI	16706052	12	8	5	3	7	7	4	3
XII	18401067	12	9	5	3	7	6	4	3
XIII	20083130	12	8	5	3	7	7	4	3
XIV	15246461	12	8	5	3	7	6	4	3
XV	16198764	12	8	5	3	7	6	4	3
XVI	18115788	12	8	5	3	7	6	4	3
XVII	14603141	12	8	5	3	7	7	4	3
XVIII	16282716	12	9	5	3	7	7	4	3
XIX	20240660	12	8	5	3	7	7	4	3
XX	19732071	12	9	5	3	7	6	4	3
XXI	11717487	12	9	5	3	7	6	4	3

Table A18: Intra-genomic variation of minimum length thresholds in the pufferfish genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
I	28185914	12	9	5	3	7	6	4	3
II	23295652	12	8	5	3	7	6	4	3
III	16798506	12	8	5	3	7	6	4	3
IV	32632948	12	9	5	3	7	6	4	3
V	12251397	12	8	5	3	7	6	4	3
VI	17083675	12	8	5	3	7	7	4	3
VII	27937443	12	8	5	3	7	7	4	3
VIII	19368704	12	8	5	3	7	6	4	3
IX	20249479	12	9	5	3	7	6	4	3
X	15657440	12	8	5	3	7	6	4	3
XI	16706052	12	8	5	3	7	7	4	3
XII	18401067	12	9	5	3	7	6	4	3
XIII	20083130	12	8	5	3	7	7	4	3
XIV	15246461	12	8	5	3	7	6	4	3
XV	16198764	12	8	5	3	7	6	4	3
XVI	18115788	12	8	5	3	7	6	4	3
XVII	14603141	12	8	5	3	7	7	4	3
XVIII	16282716	12	9	5	3	7	7	4	3
XIX	20240660	12	8	5	3	7	7	4	3
XX	19732071	12	9	5	3	7	6	4	3
XXI	11717487	12	9	5	3	7	6	4	3

Table A19: Intra-genomic variation of minimum length thresholds in the honeybee genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	29934090	13	9	5	3	13	9	5	3
2	16072177	13	9	5	3	13	9	5	3
3	13621520	13	8	5	3	13	8	5	3
4	12256690	13	9	5	3	13	9	5	3
5	14500692	13	9	5	3	13	9	5	3
6	17739083	13	9	5	3	13	9	5	3
7	12848973	13	8	5	3	13	8	5	3
8	13189223	13	9	5	3	13	9	5	3
9	11082907	13	9	5	3	13	9	5	3
10	12642577	13	9	5	3	13	9	5	3
11	14521977	13	8	5	3	13	8	5	3
12	11309010	13	8	5	3	13	8	5	3
13	10266737	12	9	5	3	12	9	5	3
14	9976661	13	9	5	3	13	9	5	3
15	10159687	13	9	5	3	13	9	5	3
16	7072872	13	8	5	3	13	8	5	3

Table A20: Intra-genomic variation of minimum length thresholds in the fruitfly genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
2L	22407834	12	10	4	3	5	9	4	3
2R	20766785	12	10	4	3	7	9	4	3
2H	1694122	13	10	5	3	7	9	4	3
3L	23771897	12	10	4	3	5	9	4	3
3R	27905053	12	10	4	3	5	9	4	3
3h	2955737	14	11	5	3	8	10	5	3
X	22224390	12	9	4	3	5	7	4	3

Table A21: Intra-genomic variation of minimum length thresholds in the mosquito (*Anopheles gambiae* PEST) genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
2L	22407834	13	9	4	3	5	9	4	3
2R	20766785	13	10	4	3	6	9	4	3
3L	23771897	14	10	4	3	6	9	4	3
3R	27905053	14	10	4	3	6	9	4	3
X	22224390	12	9	5	3	5	8	4	3

Table A22: Intra-genomic variation of minimum length thresholds in the red floor beetle genome

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
2	12900155	>20	10	7	4	6	10	6	4
3	32080666	>20	10	6	4	6	9	6	4
4	13894384	>20	10	6	3	6	9	6	3
5	18847211	>20	11	6	3	6	10	6	4
6	13544221	>20	10	6	3	6	5	5	4
7	17478683	>20	11	6	4	6	10	6	4
8	15773733	>20	10	6	3	6	8	6	4
9	15222296	>20	10	6	3	6	9	6	4
10	8806720	>20	10	6	4	6	5	6	4
X	8109244	>20	11	6	3	6	9	6	4

Table A23: Intra-genomic variation of minimum length thresholds in the roundworm genome.

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
I	15080483	>20	9	5	3	5	9	5	3
II	15279308	>20	9	5	3	5	9	5	3
III	13783313	>20	9	4	3	5	6	4	3
IV	17493791	>20	9	5	3	5	6	5	3
V	20922231	>20	9	5	3	5	9	5	3
X	17718849	>20	9	5	3	5	9	5	3

Table A24: Intra-genomic variation of minimum length thresholds in the yeast genome.

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	230208	13	NA	5	3	7		5	3
2	813178	12	NA	5	3	8	NA	5	3
3	316617	10	7	5	2	3	7	3	2
4	1531919	12	9	6	3	7	9	5	3
5	576869	10	6	2	2	2	2	2	2
6	270148	12	NA	5	3	7	NA	5	3
7	1090947	12	9	5	3	7	9	5	3
8	562643	12	8	6	3	7	8	5	3
9	439885	12	9	5	3	7	9	5	3
10	745741	12	NA	5	3	7	NA	5	3
11	666454	12	NA	5	3	7	NA	5	3
12	1078175	12	9	6	3	7	NA	5	3
13	924429	12	9	6	3	7	9	5	3
14	784333	12	NA	5	3	7	NA	5	3
15	1091289	12	9	5	3	7	9	5	3
16	948062	12	9	5	3	7	9	5	3

Table A25: Intra-genomic variation of minimum length thresholds in the *Plasmodium falciparum* genome.

		Markov				deWachter			
Chr	**Chrom. length**	A/T	C/G	DI	TRI	A/T	C/G	DI	TRI
1	643292	15	8	7	3	9	5	4	3
2	947102	15	7	8	3	9	4	4	4
3	1060087	15	6	7	3	9	4	4	3
4	1204112	15	11	7	3	9	5	4	3
5	1343552	15	7	8	3	9	4	4	3
6	1418244	15	10	7	3	9	5	4	3
7	1351552	15	7	7	3	8	4	4	3
8	1325595	15	8	8	3	9	4	4	4
9	1541723	15	6	8	3	9	4	4	4
10	1694445	15	6	8	3	9	4	4	4
11	2035250	15	7	8	3	9	4	4	4
12	2271916	15	6	8	3	9	4	4	4
13	2732359	15	8	8	3	9	4	4	4
14	3291006	15	7	7	3	8	4	4	3

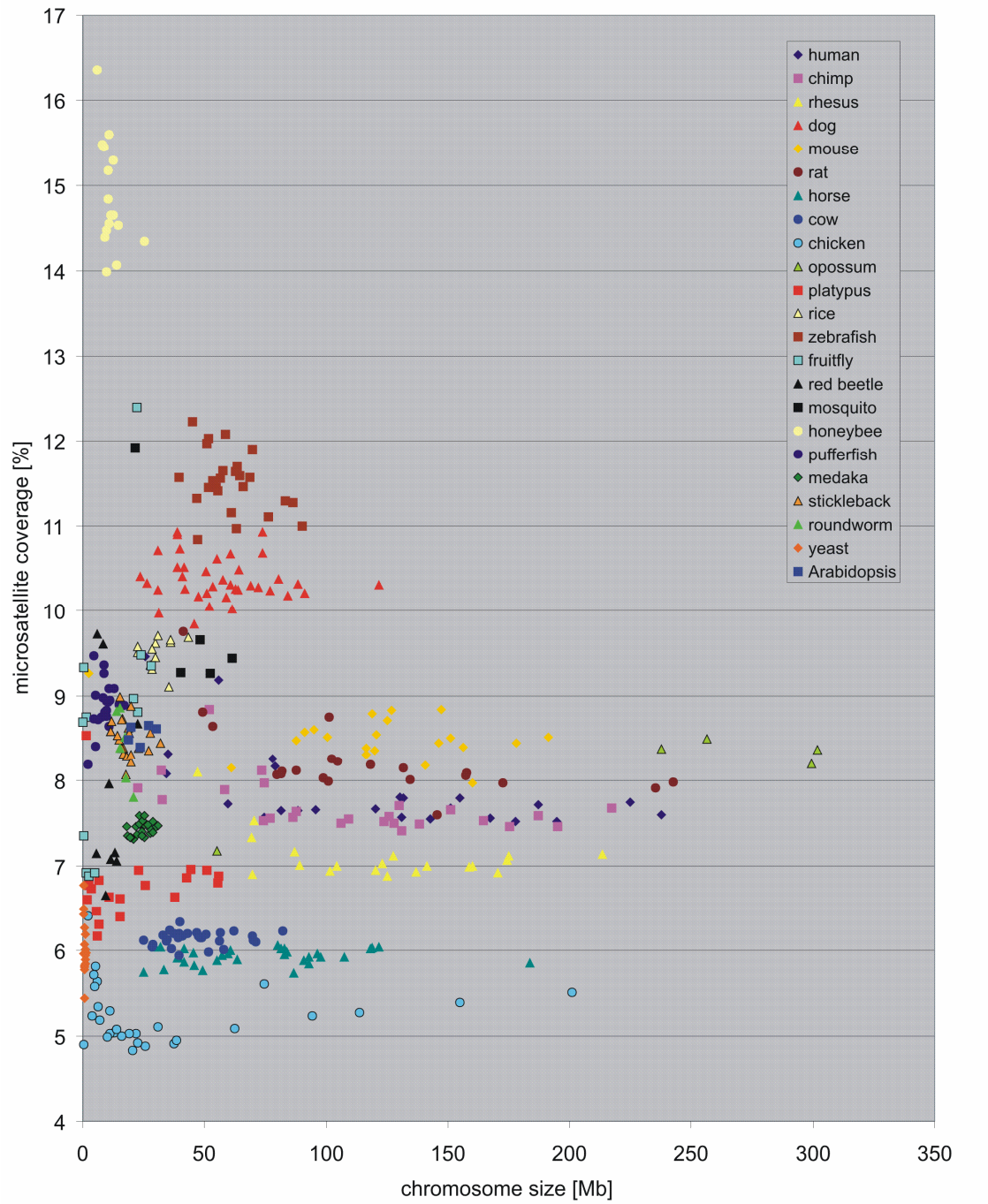

Figure A1: Percentage coverage of microsatellites in eukaryotic genomes, obtained with the program TRF (joint dataset, filtered with species-specific minimum length thresholds). The results for *Plasmodium falciparum* were left out of this graph because the microsatellite coverage in this species amounted up to 30%. The percentage coverage of microsatellites within the analyzed genomes does not depend on the total length of the sequence analyzed. Since genomic sequence length, especially in higher eukaryotes, is dependent on the content of interspersed repeats within the genome, the observed lack of correlation suggests that microsatellites are not just a consequence of interspersed repeats, but that unique genomic sequences (the sequences not corresponding to interspersed repeats, ~50% of the sequence in humans) have a higher microsatellite content than interspersed repeats.

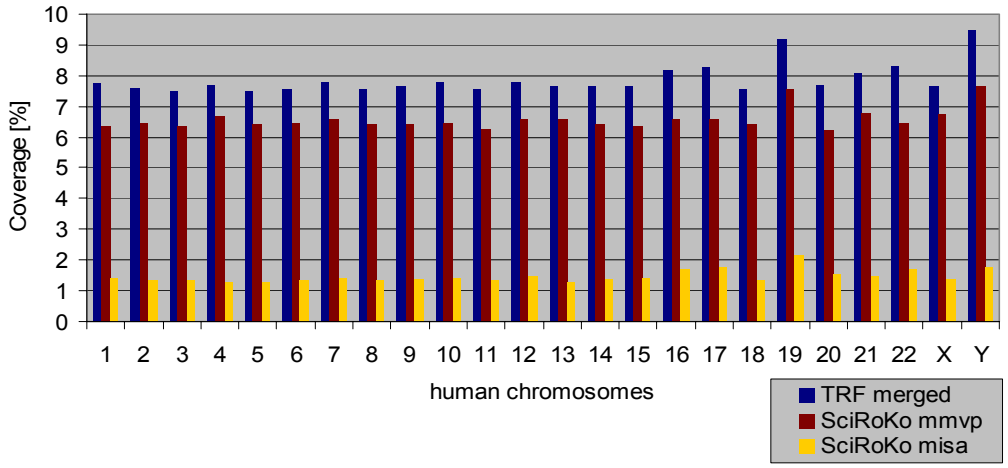

Figure A2: Intra-genomic variation of microsatellite coverage in the human genome (Hg18). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

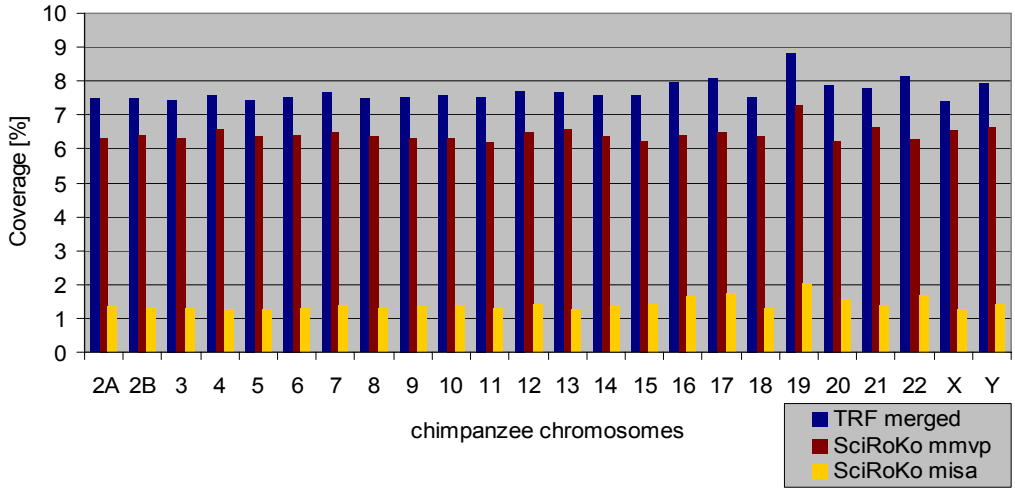

Figure A3: Intra-genomic variation of microsatellite coverage in the chimpanzee genome (panTro2). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

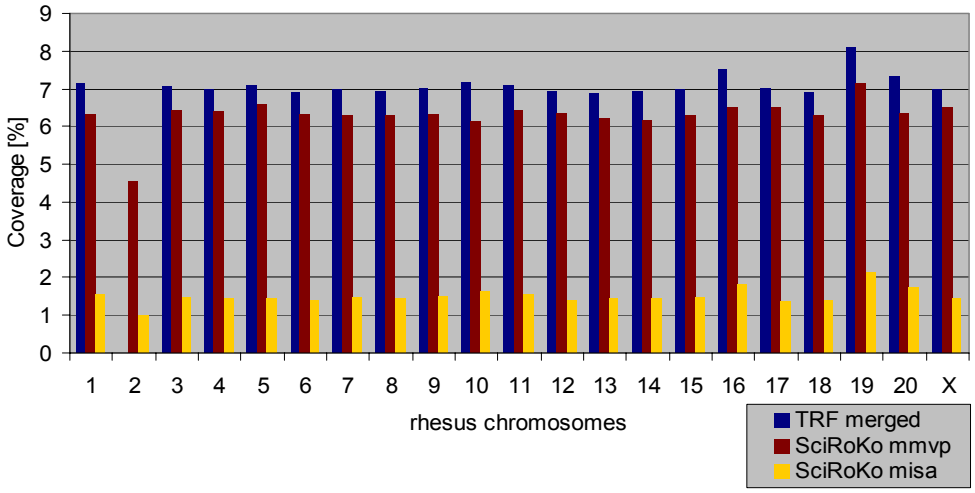

Figure A4: Intra-genomic variation of microsatellite coverage in the rhesus genome (Mmul_051212). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments. The TRF results for chromosome 2 are missing.

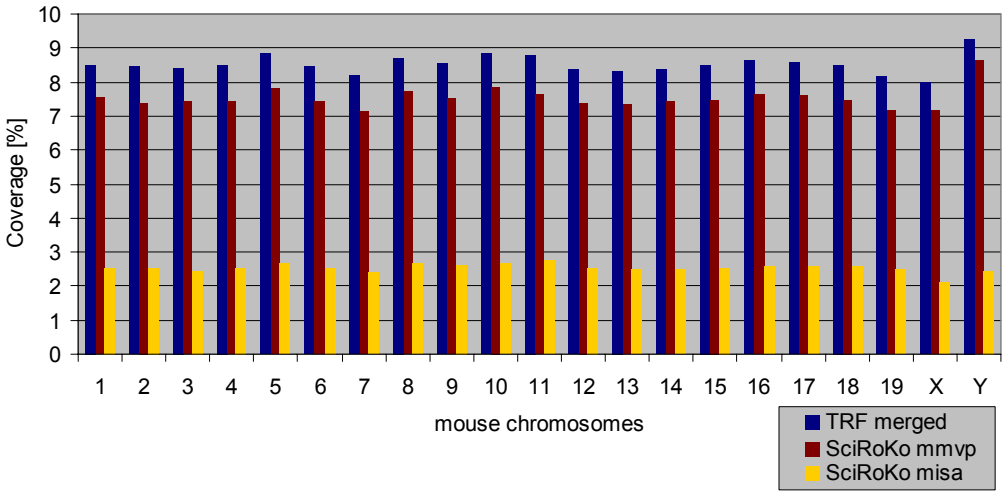

Figure A5: Intra-genomic variation of microsatellite coverage in the mouse genome (mm8). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

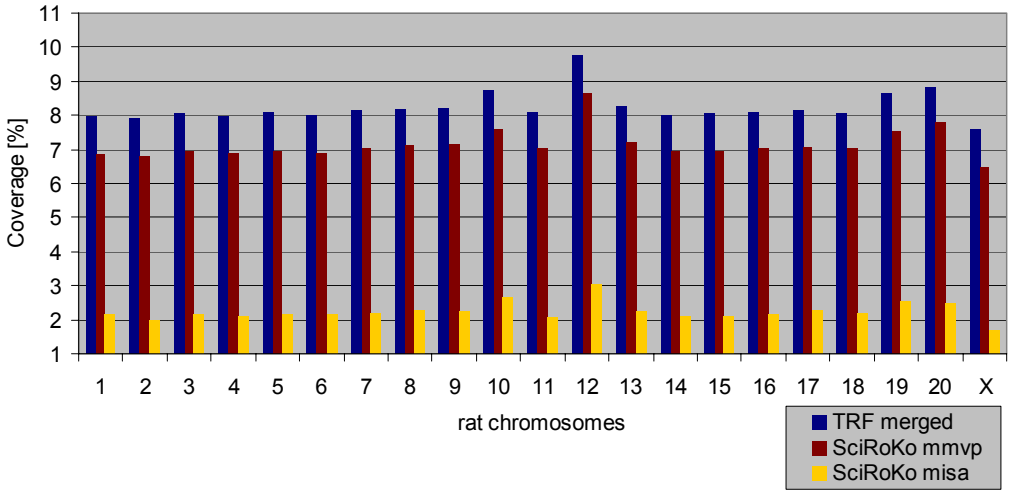

Figure A6: Intra-genomic variation of microsatellite coverage in the rat genome (rn4). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

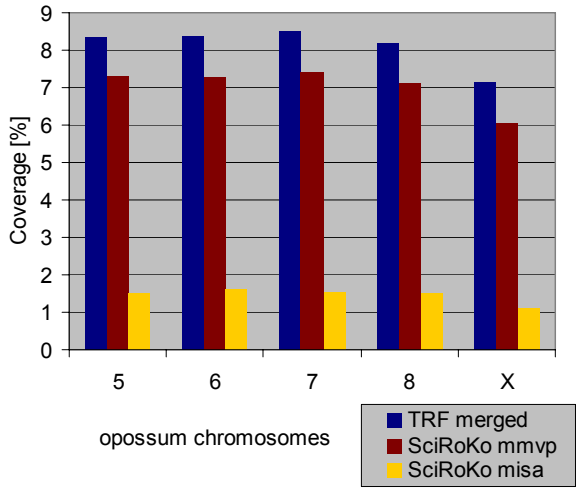

Figure A7: Intra-genomic variation of microsatellite coverage in the opossum genome (monDom4). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

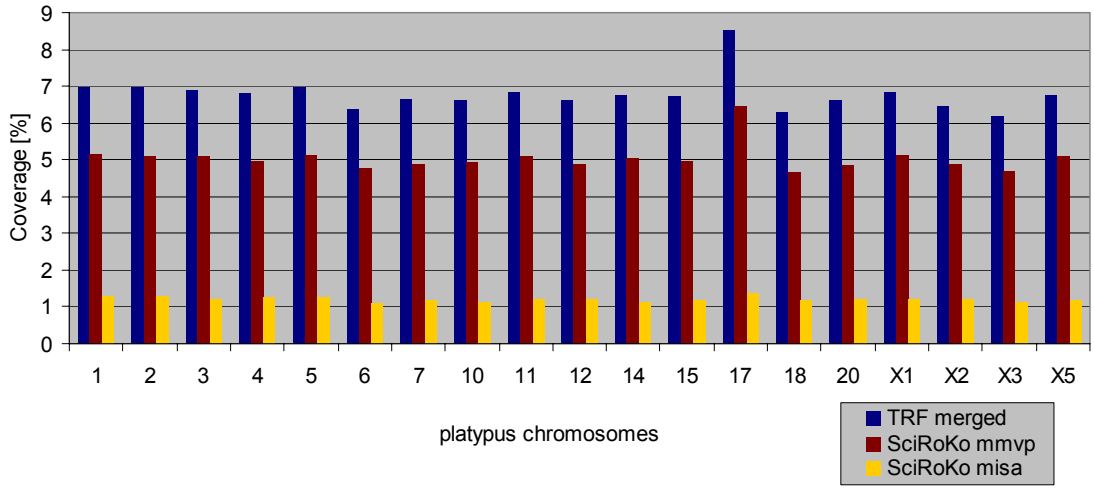

Figure A8: Intra-genomic variation of microsatellite coverage in the platypus genome (ornAna1). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

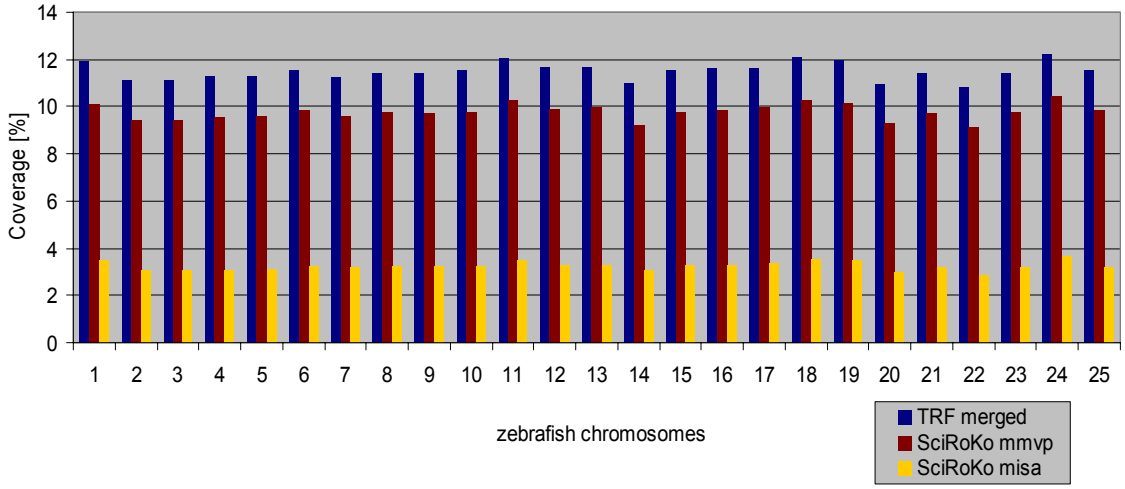

Figure A9: Intra-genomic variation of microsatellite coverage in the zebrafish genome (danRer4). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

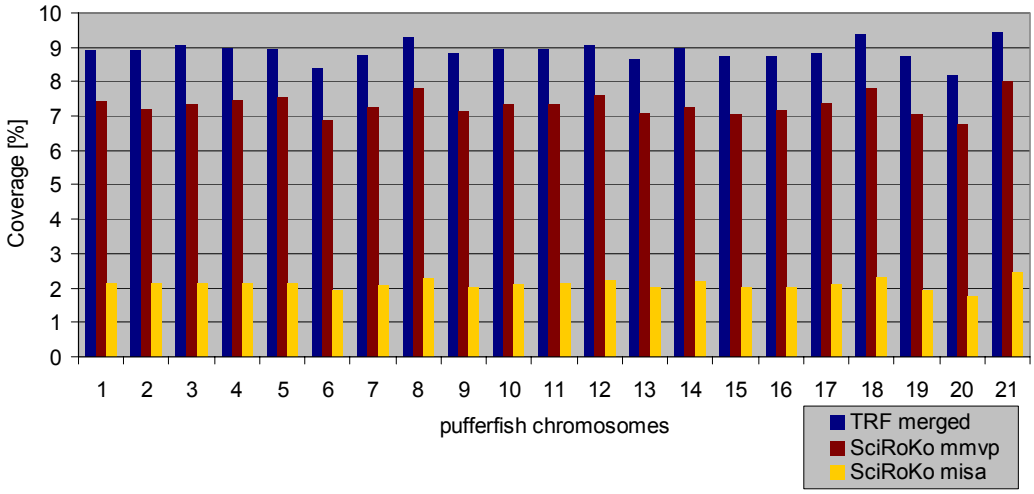

Figure A10: Intra-genomic variation of microsatellite coverage in the pufferfish genome (tetNig1). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

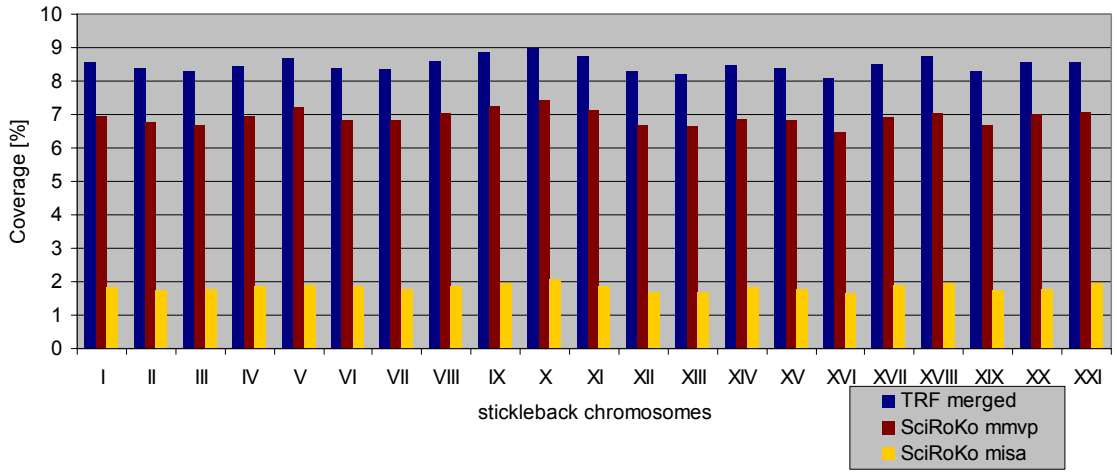

Figure A11: Intra-genomic variation of microsatellite coverage in the stickleback genome (gasAcu1). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

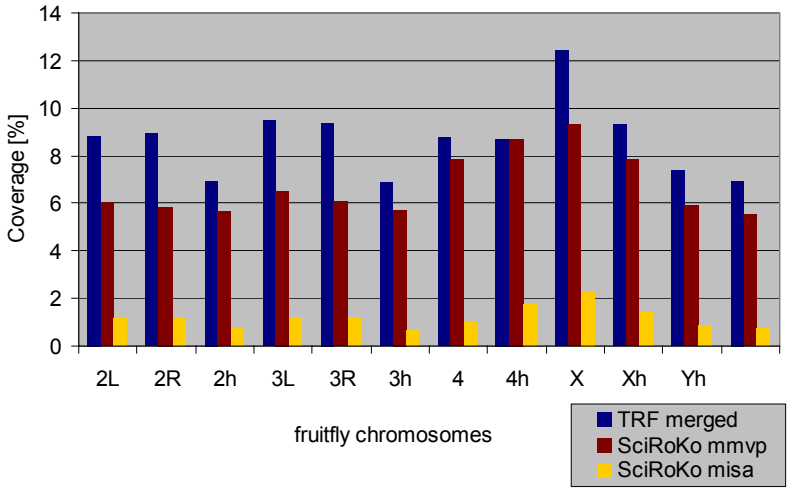

Figure A12: Intra-genomic variation of microsatellite coverage in the fruitfly genome (dm2). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

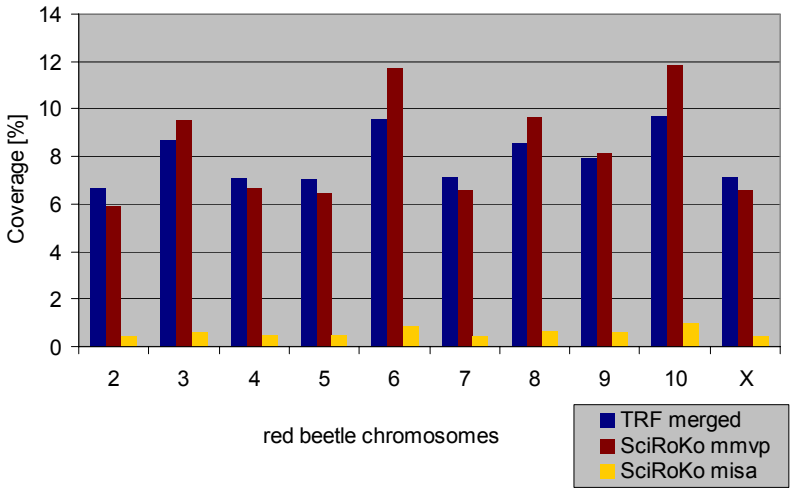

Figure A13: Intra-genomic variation of microsatellite coverage in the red beetle genome (Tcas_2.0). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

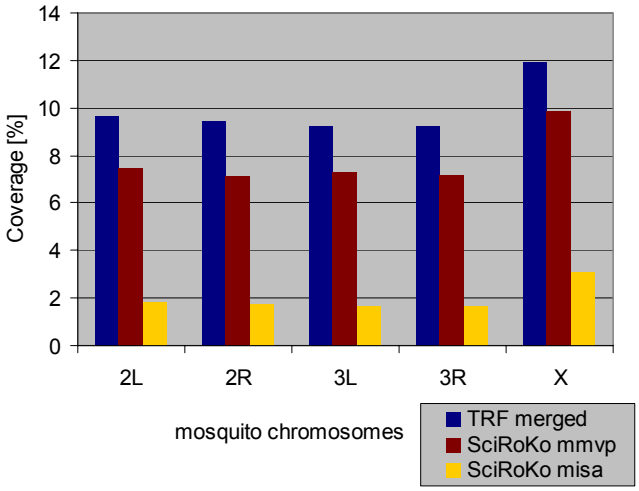

Figure A14: Intra-genomic variation of microsatellite coverage in the Anopheles mosquito genome (anoGam1). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

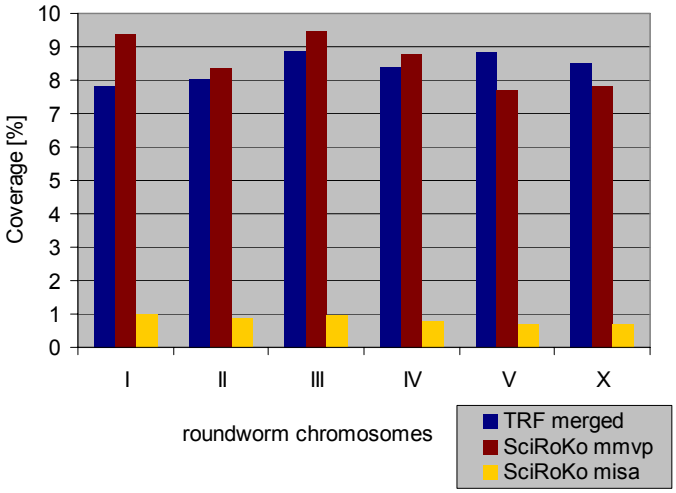

Figure A15: Intra-genomic variation of microsatellite coverage in the *C. elegans* genome (ce2). The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments.

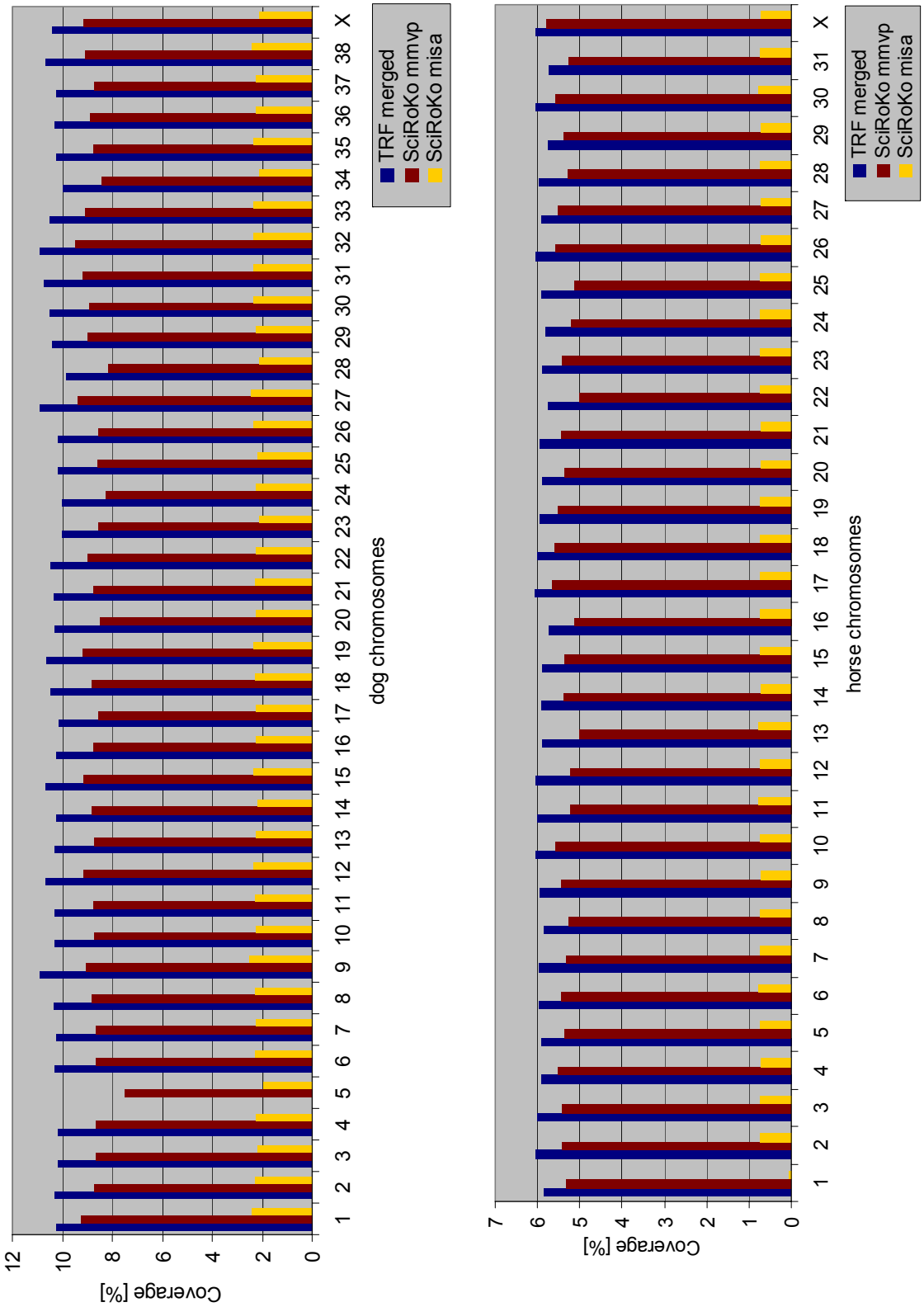

Figure A16: Intra-genomic variation of microsatellite coverage in the dog (CanFam2, left) and horse (equCab1) genomes. The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments. For the dog genome the TRF results for chromosome 5 are missing.

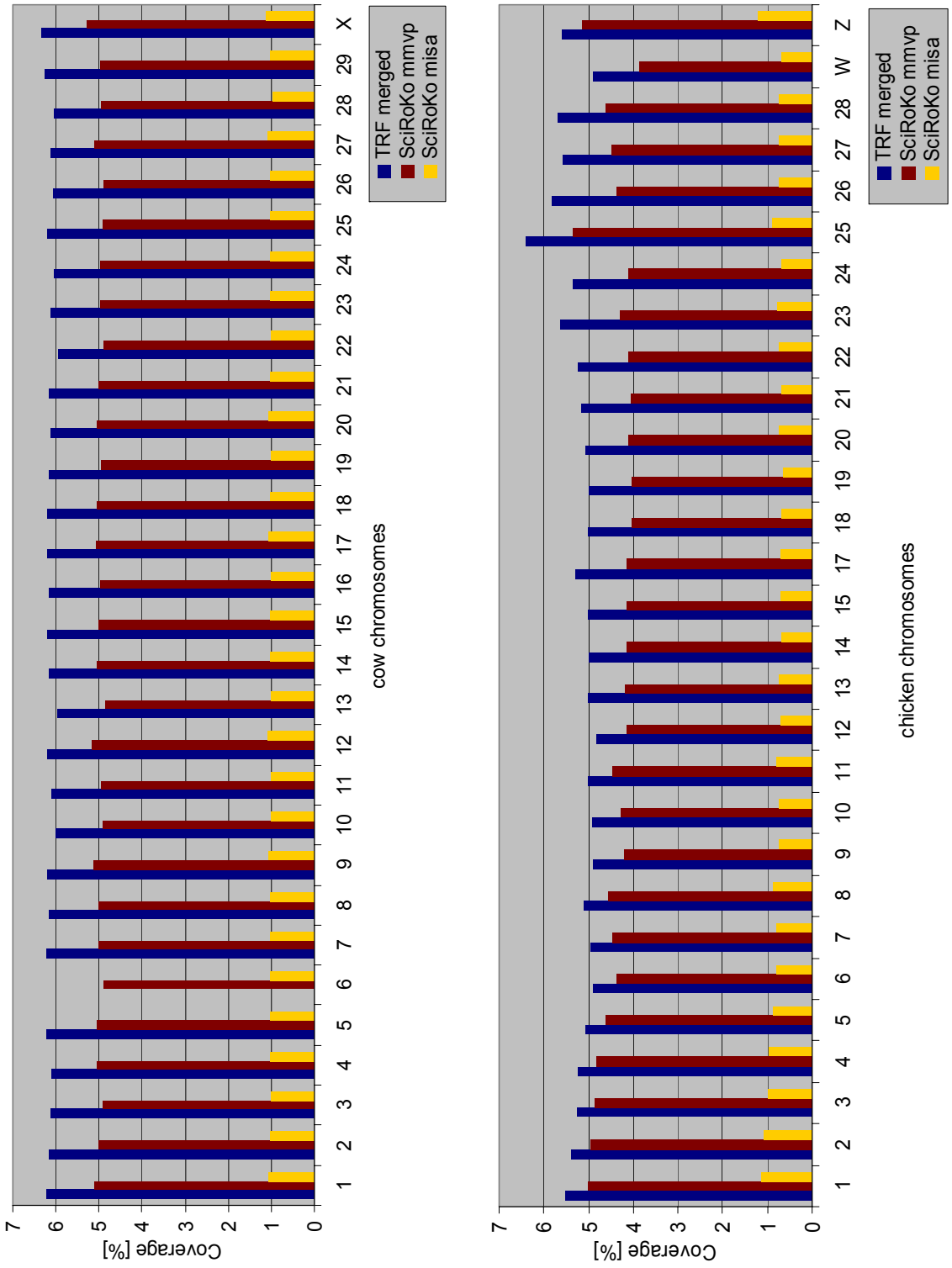

Figure A17: Intra-genomic variation of microsatellite coverage in the cow (bosTau2, left) and chicken (galGal3) genomes. The TRF and SiRoKo mmvp bars show the imperfect microsatellites, and the SciRoKo misa shows the proportion of these that are perfect tandem repeat segments. For the cow genome the TRF results for chromosome 6 are missing.

Appendix II: Supplementary Methods

Supplementary Methods

Based on the methodology designed throughout Chapters II, III, and IV, I carried out the characterization of microsatellite abundance in 24 eukaryotes, 8 prokaryotes, and 5 archaea complete and assembled genomic sequences.

Data acquisition

DNA sequences corresponding to whole assembled chromosomes were downloaded from the NCBI FTP server via the NZBioMirror (ftp://biomirror.auckland.ac.nz/ncbigenomes/) or from the FTP server from the UCSC Genome Browser (ftp://hgdownload.cse.ucsc.edu/goldenPath/).

Microsatellite searches

Microsatellites were identified in the genomes listed in section 5.3.1 using TRF and SciRoKo. All custom scripts used here were developed during the optimization of microsatellite searches in Chapter III, and the authorship is detailed in the Statement of Sources section.

TRF searches were run with two sets of parameters: 2 3 6 75 20 10 6 and 2 7 7 80 10 10 6. The resulting datasets were processed to filter out hits which were lower than two sets of minimum length thresholds, and with a percent of matches lower than 60%, using a java program specifically designed for the purpose, MsatFilter.java. One of the minimum length threshold sets used corresponds to the species-specific minimum microsatellite length thresholds based on a second order Markov model for predicting microsatellite expectations determined in Chapter IV (**table A.26**). The second minimum length threshold set corresponds to the most common minimum length threshold used in the literature: 12 nt, used in terms of repeats as 12, 6, 4 3, 3, and 3 repeats for mono- to hexanucleotide motifs respectively.

Table A.26: Species-specific minimum length thresholds based on a second order Markov model for prediction of microsatellite of microsatellite expectations.

Genome	Minimum length threshold in number of repeats			
	Mono	Di	Tri	Tetra to hexa
Homo sapiens	12	5	3	3
Pan troglodytes	12	5	3	3
Macaca mulatta	12	5	4	3
Canis familiaris	11	5	4	3
Mus musculus	11	5	4	3
Ratus norvergicus	11	5	4	3
Equus caballus	12	5	4	3
Bos taurus	12	5	4	3
Gallus gallus	11	5	4	3
Monodelphis domestica	10	5	4	3
Ornithorhynchus anatinus	11	5	3	3
Arabidopsis thaliana	11	5	3	3
Oryza sativa (japonica cultivar-group)	11	4	3	3
Danio rerio	11	5	3	3
Tetraodon nigroviridis	10	5	3	3
Oryzias latipes		5	3	3
Gasterosteus aculeatus	10	5	3	3
Apis mellifera	11	5	3	3
Anopheles gambiae str. PEST	11	4	3	3
Drosophila melanogaster	11	4	3	3
Tribolium castaneum	10*	6	3	3
Saccharomyces cerevisiae	10	5	3	3
Caenorhabditis elegans	9*	5	3	3
Plasmodium falciparum	11	8	3	3
Clostridium tetani	NA	4	NA	3
Bacillus thuringiensis	NA	2	2	3
Brucella melitensis	2	2	2	3
Lactobacillus casei	9	NA	5	3
Neisseria meningitidis	9	6	3	3
Escherichia coli K12	NA	NA	3	3
Campylobacter jejuni	3	NA	3	3
Mycobacterium tuberculosis	3	NA	NA	3
Methanosaeta thermophila	10	NA	2	3
Pyrobaculum aerophilum	12	6	3	3
Hyperthermus butylicus	NA	NA	3	3
Methanocaldococcus jannaschii	NA	NA	NA	3
Natronomonas pharaonis	2	NA	2	3

Subsequently, redundant hits were eliminated from the filtered datasets using the TRFRedundancyEliminator.java script. For this filtering process, a maximum overlap of three nnt was allowed among adjacent microsatellite hits. Tandem repeat stretches overlapping by more than 3 nt were merged into a single hit.

The search results from the two parameter sets from TRF were concatenated and merged using the Merge.java script, and total coverage of microsatellites per chromosome as well as divided into non-overlapping 100000 nt intervals was calculated using the MsatDensity.java script.

SciRoKo searches were also run with two main parameter sets, one for the identification of perfect microsatellites in MISA mode, and the other with the mismatched variable penalty mode (mmvp). The perfect search could not be carried out using the species-specific microsatellite length thresholds from **table 5.1**, because the program SciRoKo in MISA mode could not process microsatellite searches with thresholds higher than 10 repeats, as required for most mononucleotide minimum length thresholds. This is a limitation that was not noticed previously, and the problem is also present in the newest version of SciRoKo (version 3.4). The parameters for the SciroKo searches were:

SciRoKo –mode mmvp –s 8 –p 1 –seedl 6 –mmao 6

SciRoKo –mode misa –m 10 5 4 3 3 3

The SciRoKo datasets were also merged, and the coverage was calculated, using the same scripts used for the post-processing of TRF results.

Appendix III: Supplementary Results and Discussion

Supplementary Results and Discussion

Microsatellite Abundance throughout genomes

A preliminary analysis of microsatellite abundance in 24 eukaryotic, 8 prokaryotic, and 5 archaeal complete and assembled genomic sequences was carried out based on the methodology designed throughout Chapters II, III, and IV. Microsatellite content was evaluated and compared in terms of the number and percent coverage of microsatellites. Both, perfect microsatellites as well as longer microsatellites extended through imperfections, or conataining mixtures of motifs, were included in the analysis.

Two sets of minimum length threshold (MLT) were used. The first set corresponded to the most commonly used MLT in the literature; 12 nt, irrespective of motif length (see EDWARDS *et al.* 1998; GUO *et al.* 2009; JURKA and PETHIYAGODA 1995; MORGANTE *et al.* 2002; SUBRAMANIAN *et al.* 2003; TÓTH *et al.* 2000), which equates to 12, 6, 4 3, 3, and 3 repeats for mono- to hexanucleotide motifs respectively. This threshold will be referred to as the "standard MLT". The second MLT set was based on the overrepresentation thresholds obtained in Chapter IV based on a second order markov model, which were motif and species-specific (species-specific MLTs, see **table A.26** in page 200).

Based on species-specific MLTs, the total coverage of microsatellite-like regions (perfect microsatellite cores extended through imperfections allowing a maximum of 40% substitutions and indels) within all eukaryotic genomes analyzed varied from 5.06% in the chicken genome up to 31.50% in the *Plasmodium falciparum* genome (**figure A1**). In bacteria this coverage varied from 1.64% in *Lactobacillus ca*sei to 10.46% in *Campylobacter jejuni*, and in archaea coverage variation went from 1.26% in *Hyperthermus butylicus* to 8.72% in *Natronomonas pharaonis*. There were significant differences among these datasets and the ones obtained with the standard MLT among all eukaryotic species except *Saccharomyces cerevisiae*, *Tribolium castaneum*, *C. elegans,* and *Drosophila melanogaster* (paired t.test, p values: 0.1161, 0.3439, 0.2559, 0.01354, respectively). There were also differences to variable degrees among datasets obtained with species-specific MLTs and standard MLTs in prokaryotic and archaeal species, but these could not be tested statistically due to the small sample sizes (one to three

sequences per species). The differences in prokaryotes and archaea were already expected because the species-specific MLTs were very variable and much lower than the standard MLT in bacteria and archaea.

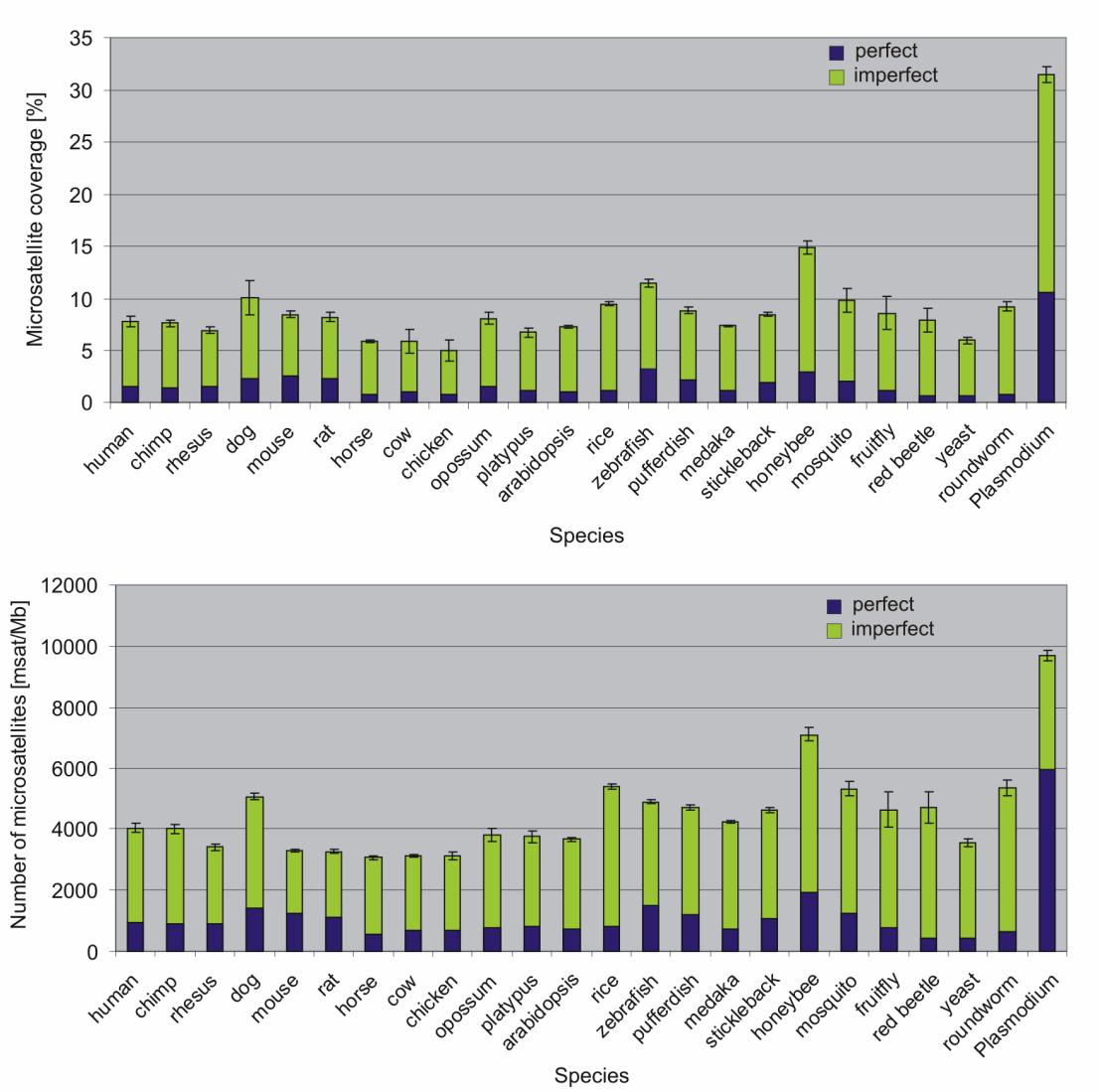

Figure A18: Total coverage (up) and numbers (down) of microsatellites detected in all eukaryotic genomes analyzed. Both of these are relative measures, so that the differences observed are directly comparable. The green part of each bar represents the imperfect part of the microsatellites reported, while the blue part represents the perfect microsatellite cores.

Relative to the percent coverage based extended imperfect microsatellites, the coverage from only perfect hits reported by the SciRoKo program in MISA mode were very low. The highest coverage among eukaryotes corresponded again to *Plasmodium falciparum* with 10.62%, and the lowest coverage was observed in the red beetle *Tribolium*

castaneum with 0.6% perfect microsatellites. Among prokaryotes and archaea, *L. casei* had the lowest coverage with 0.41% and *N. pharaonis* had the higherst coverage with 1.16%.. Bacterial genomes were usually reported to have very low microsatellite contents (COX and MIRKIN 1997; FIELD and WILLS 1996; GUR-ARIE *et al.* 2000; KASSAI-JÁGER *et al.* 2008), however, the values reported here, when expressed in microsatellites per Mb as shown in **figure A19**, are higher than the ones reported in the mentioned studies.

The abundance of perfect and imperfect microsatellites was not correlated with genome size in neither of the taxonomic groups examined, and this is in agreement with earlier studies on distinct taxonomic groups (CRUZ *et al.* 2005; EDWARDS *et al.* 1998; LIM *et al.* 2004; MORGANTE *et al.* 2002). The global content of imperfection in microsatellites (ratio imperfect/perfect, **figure A20**) had also no relationship with genome size, which is evident by the contents of imperfect microsatellites in bacteria and archaea (**figure A19**), which are in the same range as the ones for eukaryotes (**figure A18**). Most of the perfect microsatellite hits reported here were immersed within longer imperfect hits, as could be seen by comparison of coordinates among perfect and imperfect datasets. This was the case in all eukaryotes, while in prokaryotes and archaea the exceptions were *L. casei*, and *H. butylicus*, *M. tuberculosis*. These three genomes contain mostly trinucleotide repeats, which tend to be immersed within coding sequences, and this could be a reason for the lack of imperfections in these repeats (BAYLISS *et al.* 2004). A study by Kassai-Jáger *et al.* (2008), the only study which analyzed imperfect microsatellites in prokaryotes (*E. coli* and Clamydial strains), also found that most perfect microsatellite hits were immersed within imperfect hits, and that the distribution of imperfect and perfect repeats followed very similar distributions. This is, however, in contraposition to findings from Metzgar *et al.* (2002), because imperfect microsatellite-like regions surrounding perfect microsatellites often indicate past expansion and point mutation events. Metzgar *et al.* (2002) suggested that microsatellites in bacteria are subject to deletion bias, and should be driven to extinction by mutational pressure whenever they are not maintained by selection. This would imply that, in bacteria, microsatellites do not expand and subsequently get interrupted by point mutations, as is believed to be the case in eukaryotes (see BULL *et al.* 1999; ELLEGREN 2004), producing degraded microsatellites around them. From the results obtained here, all genomes analyzed, contained imperfect microsatellites to some degree, which suggests that overall expansion and subsequent generation of imperfect and

composite microsatellties through point mutations is a common process in microsatellite evolution among all taxa.

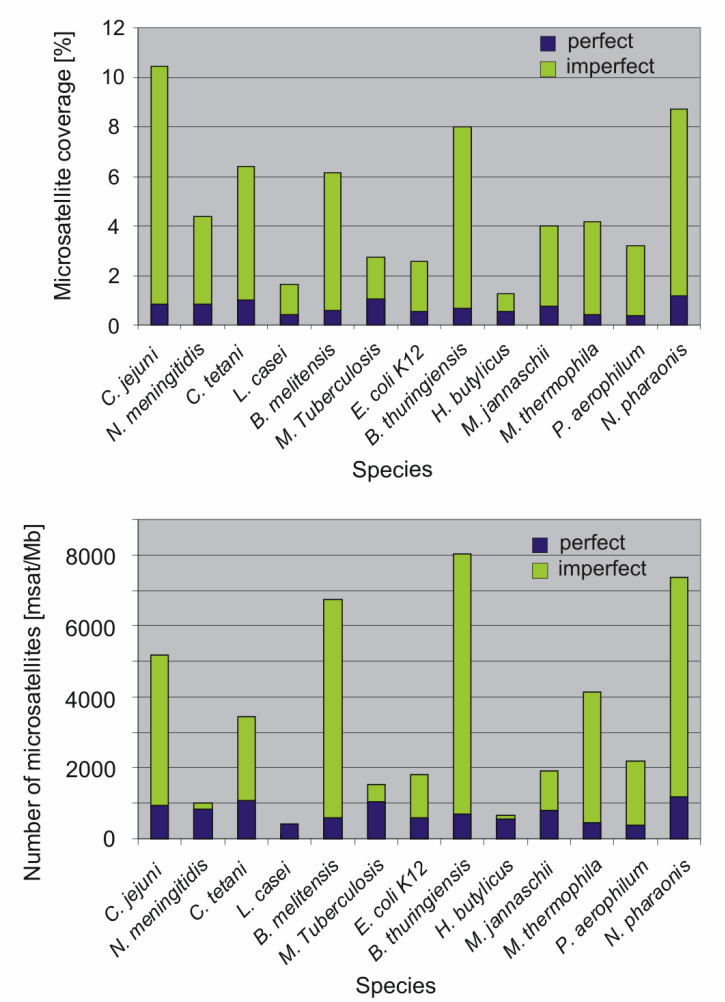

Figure A19: Total coverage (up) and numbers (down) of microsatellites detected in prokaryotic and archaeal (last five to the left) genomes. Both of these are relative measures, so that the differences observed are directly comparable. The green part of each bar represents the imperfect part of the microsatellites reported, while the blue part represents the perfect microsatellite cores.

In contrast to the large inter-genome variation in microsatellite content, microsatellite abundance among chromosomes within the same genome remained relatively constant (see **figures A2** to **A17** in Appendix I). The ratios of imperfect to perfect microsatellites were also remarkably constant among chromosomes from the same genome. This observation suggests that, despite of the heterogeneity of microsatellite dynamics based on mutation rates reported for microsatellites with different motifs (ANDERSON *et al.* 2000; KELKAR *et al.* 2008; PRIMMER and ELLEGREN 1998), lengths (BROHEDE *et al.* 2002; ELLEGREN

2000), genomic positions (METZGAR *et al.* 2000), individual-specific mutation variations (e.g. BROHEDE *et al.* 2004), etc., the main factors governing microsatellite evolution are genome-wide and species-specific mechanisms like sequence replication and repair machineries.

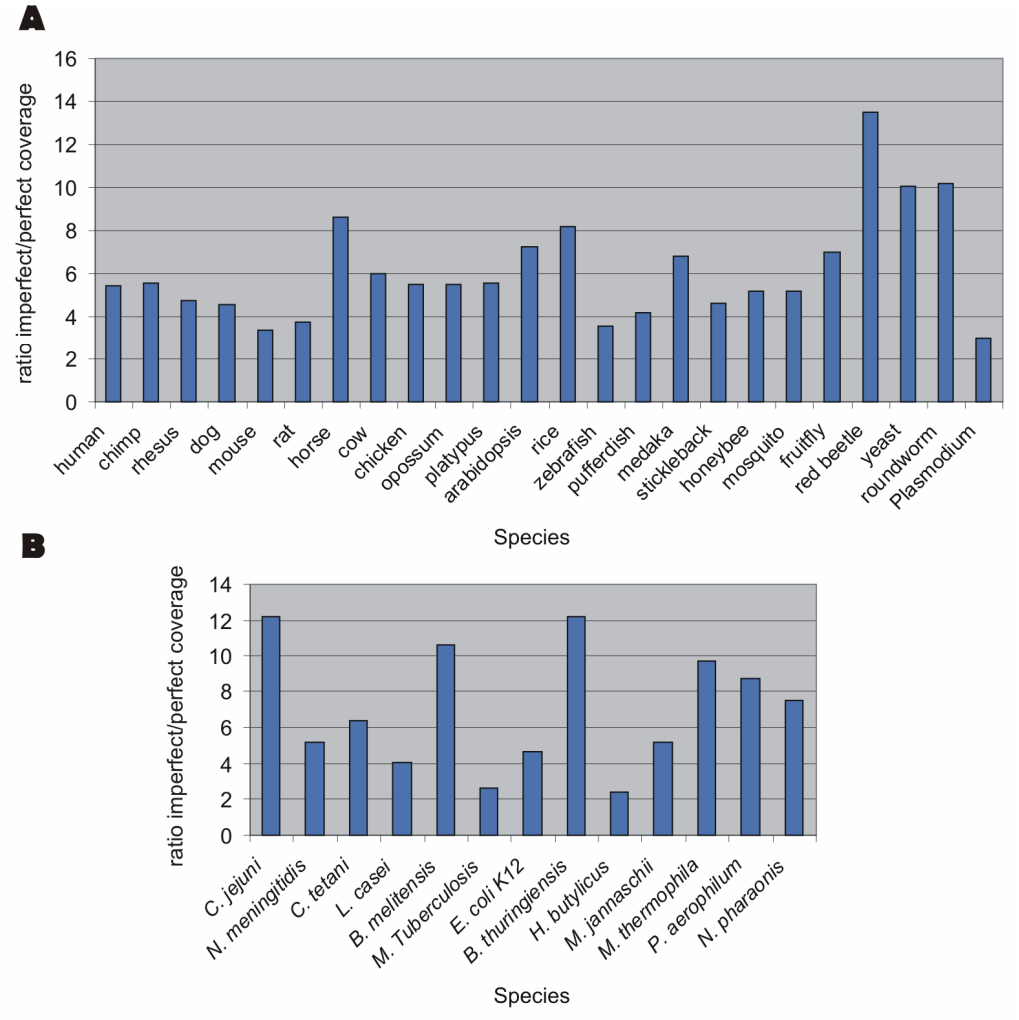

Figure A20: Ratios of imperfect to perfect microsatellite hits reported in eukaryotes (A), prokaryotes and archaea (B). The higher this ratio, the more imperfect microsatellites are part of the respective genomes.

The global microsatellite abundance data presented here is an attempt at analyzing microsatellite content based on species-specific MLTs. Within each species, a mode MLT was used for every motif size class, therefore grouping all motifs by motif size regardless of motif nucleotide composition. However, it is important to notice that there were also

differences in expectations, and therefore inMLTs, among motifs with different nucleotide composition. Motifs containing CG dinucleotides were not abundant in the majority of eukaryotes, and did not reach overrepresentation in most prokaryotes and archea. Therefore, in particular in prokaryotes and archaea, it would be recommented to use also motif-specific MLTs. This would imply doing independent searches for each microsatellite motif, to further join and filter the datasets for reduncancy afterwards. However, given the small genome sizes of prokaryotes and archaea, this is a relatively feasible endeavor which would be worth undertaking.

References

ANDERSON, T. J. C., X.-Z. SU, A. RODDAM and K. P. DAY, 2000 Complex mutations in a high proportion of microsatellite loci from the protozoan parasite *Plasmodium falciparum*. Mol Ecol **9:** 1599-1608.

BAYLISS, C. D., K. M. DIXON and E. R. MOXON, 2004 Simple sequence repeats (microsatellites): mutational mechanisms and contributions to bacterial pathogenesis. A meeting review. FEMS Immunol Med Microbiol **40:** 11-19.

BROHEDE, J., A. P. MOLLER and H. ELLEGREN, 2004 Individual variation in microsatellite mutation rate in barn swallows. Mutat Res **545:** 73-80.

BROHEDE, J., C. R. PRIMMER, A. MOLLER and H. ELLEGREN, 2002 Heterogeneity in the rate and pattern of germline mutation at individual microsatellite loci. Nucleic Acids Res **30:** 1997-2003.

BULL, L. N., C. R. PABON-PENA and N. B. FREIMER, 1999 Compound microsatellite repeats: practical and theoretical features. Genome Res **9:** 830-838.

COX, R., and S. M. MIRKIN, 1997 Characteristic enrichment of DNA repeats in different genomes. Proc Natl Acad Sci U S A **94:** 5237-5242.

CRUZ, F., M. PEREZ and P. PRESA, 2005 Distribution and abundance of microsatellites in the genome of bivalves. Gene **346:** 241-247.

EDWARDS, Y. J., G. ELGAR, M. S. CLARK and M. J. BISHOP, 1998 The identification and characterization of microsatellites in the compact genome of the Japanese pufferfish, *Fugu rubripes*: perspectives in functional and comparative genomic analyses. J Mol Biol **278:** 843-854.

ELLEGREN, H., 2000 Heterogeneous mutation processes in human microsatellite DNA sequences. Nat Genet **24:** 400-402.

ELLEGREN, H., 2004 Microsatellites: simple sequences with complex evolution. Nat Rev Genet **5:** 435-445.

FIELD, D., and C. WILLS, 1996 Long, polymorphic microsatellites in simple organisms. Proceedings of the Royal Society B: Biological Sciences **263:** 209-215.

GUO, W. J., J. LING and P. LI, 2009 Consensus features of microsatellite distribution: Microsatellite contents are universally correlated with recombination rates and are preferentially depressed by centromeres in multicellular eukaryotic genomes. Genomics.

GUR-ARIE, R., C. J. COHEN, Y. EITAN, L. SHELEF, E. M. HALLERMAN *et al.*, 2000 Simple sequence repeats in *Escherichia coli*: abundance, distribution, composition, and polymorphism. Genome Res **10:** 62-71.

JURKA, J., and C. PETHIYAGODA, 1995 Simple repetitive DNA sequences from primates: compilation and analysis. J Mol Evol **40:** 120-126.

KASSAI-JÁGER, E., C. ORTUTAY, G. TÓTH, T. VELLAI and Z. GASPARI, 2008 Distribution and evolution of short tandem repeats in closely related bacterial genomes. Gene **410:** 18-25.

KELKAR, Y. D., S. TYEKUCHEVA, F. CHIAROMONTE and K. D. MAKOVA, 2008 The genome-wide determinants of human and chimpanzee microsatellite evolution. Genome Res **18:** 30-38.

LIM, S., L. NOTLEY-MCROBB, M. LIM and D. A. CARTER, 2004 A comparison of the nature and abundance of microsatellites in 14 fungal genomes. Fungal Genet Biol **41:** 1025-1036.

METZGAR, D., J. BYTOF and C. WILLS, 2000 Selection against frameshift mutations limits microsatellite expansion in coding DNA. Genome Res **10:** 72-80.

METZGAR, D., L. LIU, C. HANSEN, K. DYBVIG and C. WILLS, 2002 Domain-level differences in microsatellite distribution and content result from different relative rates of insertion and deletion mutations. Genome Res **12:** 408-413.

MORGANTE, M., M. HANAFEY and W. POWELL, 2002 Microsatellites are preferentially associated with nonrepetitive DNA in plant genomes. Nat Genet **30:** 194-200.

PRIMMER, C. R., and H. ELLEGREN, 1998 Patterns of molecular evolution in avian microsatellites. Molecular Biology and Evolution **15:** 997-1008.

SUBRAMANIAN, S., R. K. MISHRA and L. SINGH, 2003 Genome-wide analysis of microsatellite repeats in humans: their abundance and density in specific genomic regions. Genome Biol **4:** R13.

TÓTH, G., Z. GASPARI and J. JURKA, 2000 Microsatellites in different eukaryotic genomes: survey and analysis. Genome Res. **10:** 967-981.

Appendix IV: Publications

Publication 1

An earlier version of Chapter I was published as an introductory review on microsatellite evolution in the Encyclopedia of Life Sciences (ELS) published by Wiley Interscience (attached):

VARGAS JENTZSCH, I. M., A. BAGSHAW, E. BUSCHIAZZO, A. MERKEL and N. J. GEMMELL, 2008 Evolution of microsatellite DNA in *Encyclopedia of Life Sciences*. John Wiley & Sons, Ltd., Chichester.

Publication 2

WARREN, W. C., L. W. HILLIER, J. A. MARSHALL GRAVES, E. BIRNEY, C. P. PONTING *et al.*, 2008 Genome analysis of the platypus reveals unique signatures of evolution. Nature **453:** 175-183.

I contributed to the analysis of the platypus genome sequence by carrying out the microsatellite abundance and distribution analysis. For this analysis I used an earlier version of my microsatellite search methodology, involving the combination of TRF (BENSON 1999) and Sputnik (ABAJIAN 1994) search results (the program SciRoKo (KOFLER *et al.* 2007), which I use in combination with TRF throughout my PhD thesis was not yet published at the time). The minimum length threshold used for these searches was set at 15 nt to make sure only long overrepresented microsatellites were reported.

Analysis of microsatellites in the platypus genome

Introduction

Microsatellites, or simple sequence repeats (SSRs), are highly polymorphic DNA sequences that consist of short (1–6 bp) tandemly repeated motifs (BUSCHIAZZO and GEMMELL 2006). Microsatellites are distributed abundantly throughout the genomes of most organisms, but their function and evolution are not yet well understood. However, whole genome sequence analyses and comparisons enable the rapid identification of

microsatellites and other genomic elements that are, or are not, conserved across evolutionary time; data that may provide important clues about their functions (KASHI and KING 2006; LI *et al.* 2002; TOTH *et al.* 2000). Here, we analyzed the abundance and distribution of microsatellites in the platypus genome and compared these to three representative mammalian genomes (human, mouse, and opossum) and two non-mammalian vertebrates (chicken and the Anole lizard (*Anolis carolinensis*)). In addition we examined the extent of conservation of microsatellite loci in the multiple alignment of these genomes to that of the platypus.

Methodology

Microsatellites were identified across the platypus genome (ornAna1) combining two programs: Tandem Repeat Finder Tandem Repeat Finder (BENSON 1999) and a modification to Sputnik (LA ROTA 2003). The assembled chromosomes and the unassembled contigs and ultracontigs were analyzed separately. Microsatellites are usually defined as perfect repetitions of 1-6 nucleotide motifs, but can also be interrupted by point mutations or possess a mixture of different motifs within the same locus. For our analyses the minimum length for a microsatellite was set to fifteen nucleotides and independent searches were run with stringent and relaxed parameters to find perfect and imperfect microsatellites, respectively. The parameters for each program were as follows. TRF: perfect repeats (2, 7, 7, 80, 10, 30, 6); imperfect repeats (2, 3, 5, 80, 10, 30, 6). Sputnik: perfect repeats (-v 1 -u 5 -s 11 -p -r 0 -L 15 -l -1); Long imperfect repeats (-v 4 -u 5 -m 2 -n -6 -s 24 -A -p -L 16 -l -1); Short imperfect repeats (-v 1 -u 3 -m 2 -n -6 -s 16 -A -p -L 16 -l -1). Under the chosen parameters, using both Sputnik and TRF, improves search sensitivity by an additional 10% over searches employing TRF alone (Vargas *et al.*, unpublished).

Inter- and intra-genomic comparisons were carried out with microsatellite datasets for human (hg18), dog (canFam2), mouse (mm8), opossum (monDom4), chicken (galGal3) and lizard (anoCar1) genomes (obtained from the UCSC Genome Browser) using the parameters described above (VARGAS *et al.* unpublished). To obtain a thorough but non-redundant estimate of microsatellite density the results for perfect and imperfect microsatellites were merged and analyzed using Visual Basic and Java scripts. Density, shown as percentage coverage, was calculated as the total length of microsatellites for

each 10kb non-overlapping window of genomic sequence. Only windows uninterrupted by gaps were analyzed. The density plots from these analyses are available on request.

The characterization of microsatellites in terms of array length, AT-content, motif and size class preference were performed using just TRF data for all genomes analysed but for dog. The set was filtered for redundant microsatellites (i.e. those occupying the same genomic position) retaining the longest array and for microsatellites with more than 3 repeats using Visual Basic scripts. The subsequent analyses were done using R, version 2.2.1 (http://www.R-project.org).

Results and Discussion

Microsatellites were identified by combining two repeat finding programs: Tandem Repeat Finder (BENSON 1999) and a modification to Sputnik (LA ROTA 2003). The analysis was performed taking into account both perfect and imperfect repeats (see supplementary methods). The platypus genomic sequences assembled into chromosomes have a mean microsatellite coverage of 2.67% (SD ±0.34), which is significantly lower than all other mammalian genomes sequenced to date and most similar to that observed in chicken (Fig. 1). Microsatellite coverage in the lizard genome is also in the realm of 3%, but this genome is not assembled into chromosomes, thus several scaffolds could be redundant and coverage overestimated. In fact, platypus has the lowest microsatellite coverage of all six species examined, due mainly to microsatellites being on average shorter in platypus than in other genomes (**Fig. 2**). However it surpasses chicken in the coverage plot (**Fig. 1**) because of an abundance of long tri- and tetranucleotide repeats. Surprisingly, the mono- and dinucleotide repeats common to the other mammalian genomes are exceedingly rare in the platypus genome. This may be a consequence of difficulties in sequencing this fraction of the platypus genome (although one must ask why this genomic fraction has proven intractable in this instance), or may reflect a genuine composition difference among the genomes, strengthening further the similarities between platypus and the chicken and lizard genomes. This latter view is supported by analyses of motif preference, with the motif AAT particularly common in the genomes of platypus, lizard, and chicken. Differences in microsatellite content observed among genomes are not due to genome size differences, because genome sizes (in terms of nucleotides of available sequence

data) do not correlate with either microsatellite coverage or number (**Fig. 2**) (VARGAS *et al.*, unpublished).

At a chromosome specific level the distribution of microsatellites is rather homogeneous with a non-significant, but positive trend between microsatellite coverage and chromosome size observed in all species. However, in both chicken and platypus, microsatellites are overrepresented in several of the small chromosomes, occurring at densities two-fold greater than the densities observed in the other chromosomes in these genomes. This similarity between chicken and platypus, together with the overall similarity in microsatellite abundance, suggests that there may be some homology between the avian microchromosomes and the small chromosomes found in platypus, such as chromosome 17.

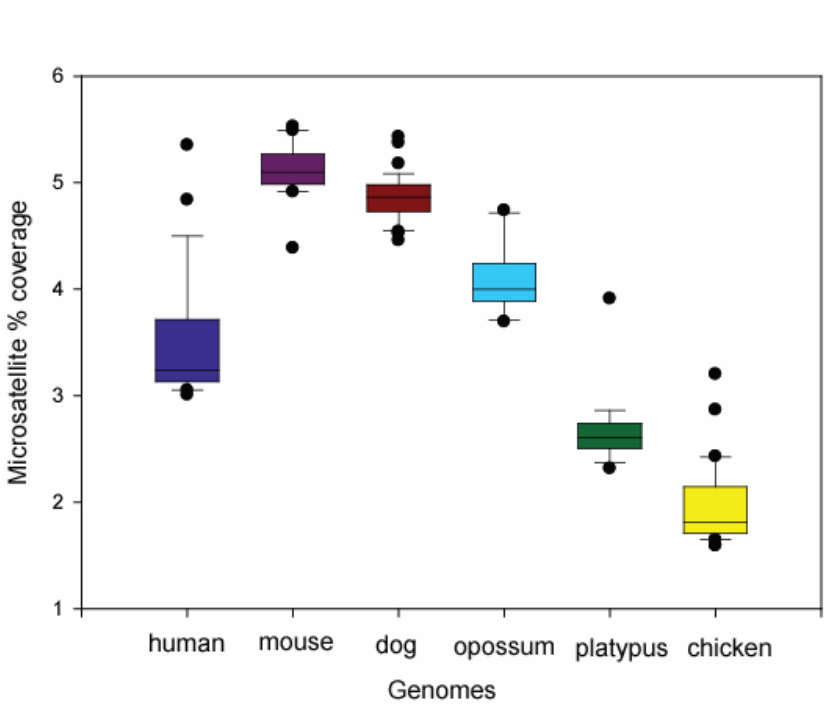

Figure 1: Whole genome microsatellite coverage compared across representative mammalian and avian genomes. For each species, the variation in microsatellite coverage by chromosome is represented by the box plot.

The platypus G+C nucleotide composition can be 2 to 8% higher than that found in eutherians and metatherians and resembles that of the chicken genome (**Fig. 3**). However, microsatellite nucleotide composition differs from the overall genomic values in all genomes analysed, with microsatellites having a higher A+T content in both perfect and

imperfect repeats, and G+C microsatellites being exceedingly rare in all cases. The analysis of microsatellite abundance by A+T content identifies four major peaks, corresponding to approximately 0, 50, 70 and 90% A+T. In comparison to the other mammals, platypus has fewer microsatellites with ~50% A+T and more with ~70% A+T, leading to an abundance distribution that has slightly more in common with chicken and lizard than with mammals.

The heightened abundance of A+T in the microsatellite sequences of the G+C rich platypus and chicken genomes suggests that the microsatellite repeats emerge from a process that is non-random and independent of genomic sequence composition. Most genomes also show variation in nucleotide composition between perfect and imperfect microsatellite fractions, suggesting that microsatellite sequences become more G+C rich as they accumulate point mutations with time (**Fig. 3**).

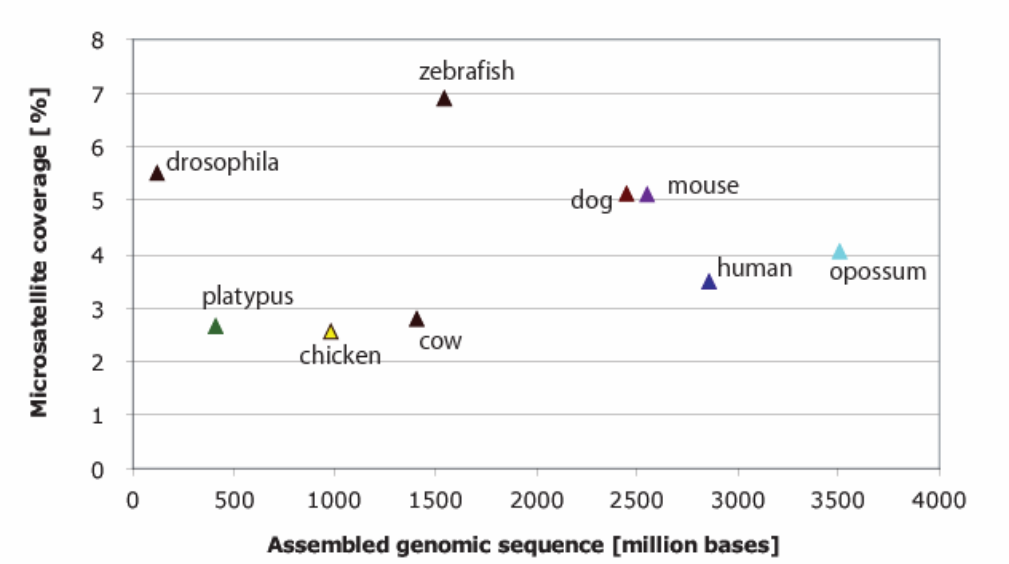

Figure 2: Plot of microsatellite coverage vs assembled genome sequence. No correlation can be observed across a wide range of genome sizes. For clarity, data from fruitfly, zebrafish and cow are added to this graph in addition to the genomes used in the platypus genomic comparison.

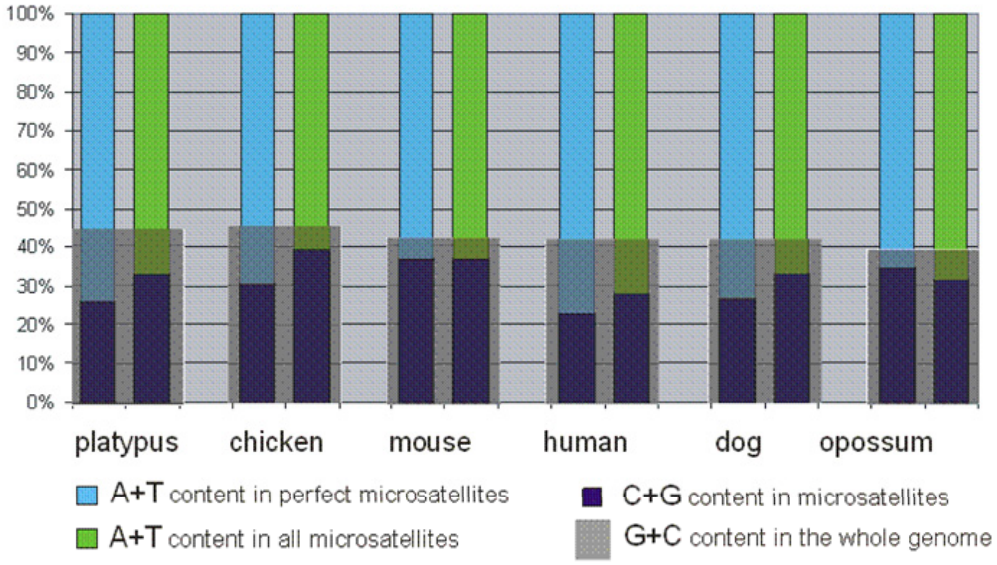

Figure 3: Pattern of nucleotide composition in perfect and imperfect microsatellites. The overall G+C content for each genome is shaded in grey. In all but the opossum genome the G+C content increases as microsatellites gain point mutations and become imperfect repeats.

References

ABAJIAN, C., 1994 Sputnik, pp. Program for finding microsatellites, Washington.

BENSON, G., 1999 Tandem repeats finder: a program to analyze DNA sequences. Nucleic Acids Res **27:** 573-580.

BUSCHIAZZO, E., and N. J. GEMMELL, 2006 The rise, fall and renaissance of microsatellites in eukaryotic genomes. Bioessays **28:** 1040-1050.

KASHI, Y., and D. G. KING, 2006 Simple sequence repeats as advantageous mutators in evolution. Trends Genet **22:** 253-259.

KOFLER, R., C. SCHLÖTTERER and T. LELLEY, 2007 SciRoKo: A new tool for whole genome microsatellite search and investigation. Bioinformatics **23:** 1683-1685.

LA ROTA, M., 2003 Sputnik, pp. Modified version of Sputnik.

LI, Y. C., A. B. KOROL, T. FAHIMA, A. BEILES and E. NEVO, 2002 Microsatellites: genomic distribution, putative functions and mutational mechanisms: a review. Mol Ecol **11:** 2453-2465.

TOTH, G., Z. GASPARI and J. JURKA, 2000 Microsatellites in different eukaryotic genomes: survey and analysis. Genome Res. **10:** 967-981.